Compression of Numerical Data-sets with the BigWhoop Library

Patrick Vogler - patrick.vogler@hlrs.de
High Performance Computing Centre
Stuttgart

Motivation

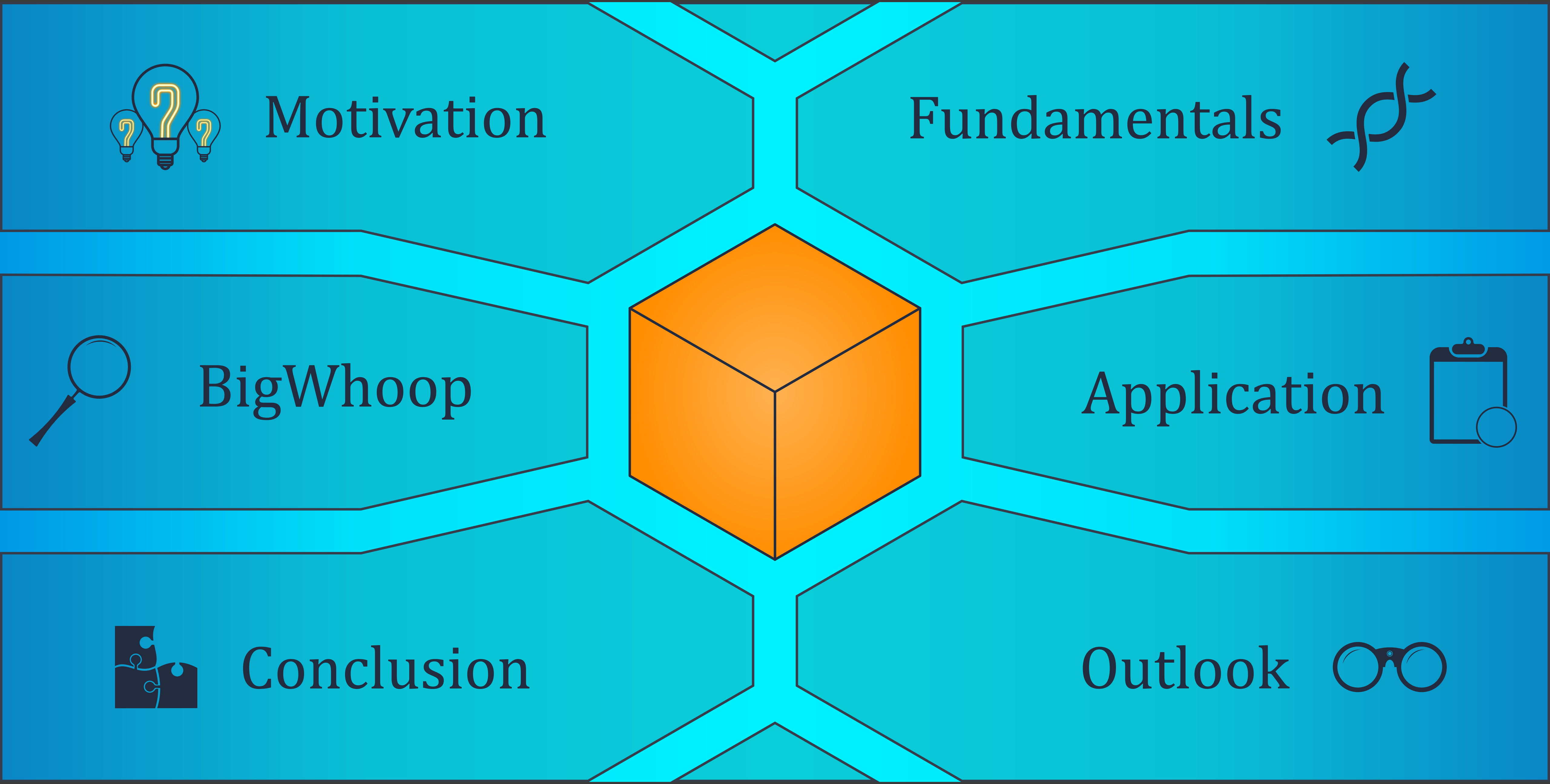
Fundamentals

BigWhoop

Application

Conclusion

Outlook


Motivation
Fundamentals
BigWhoop
Application
Conclusion
Outlook

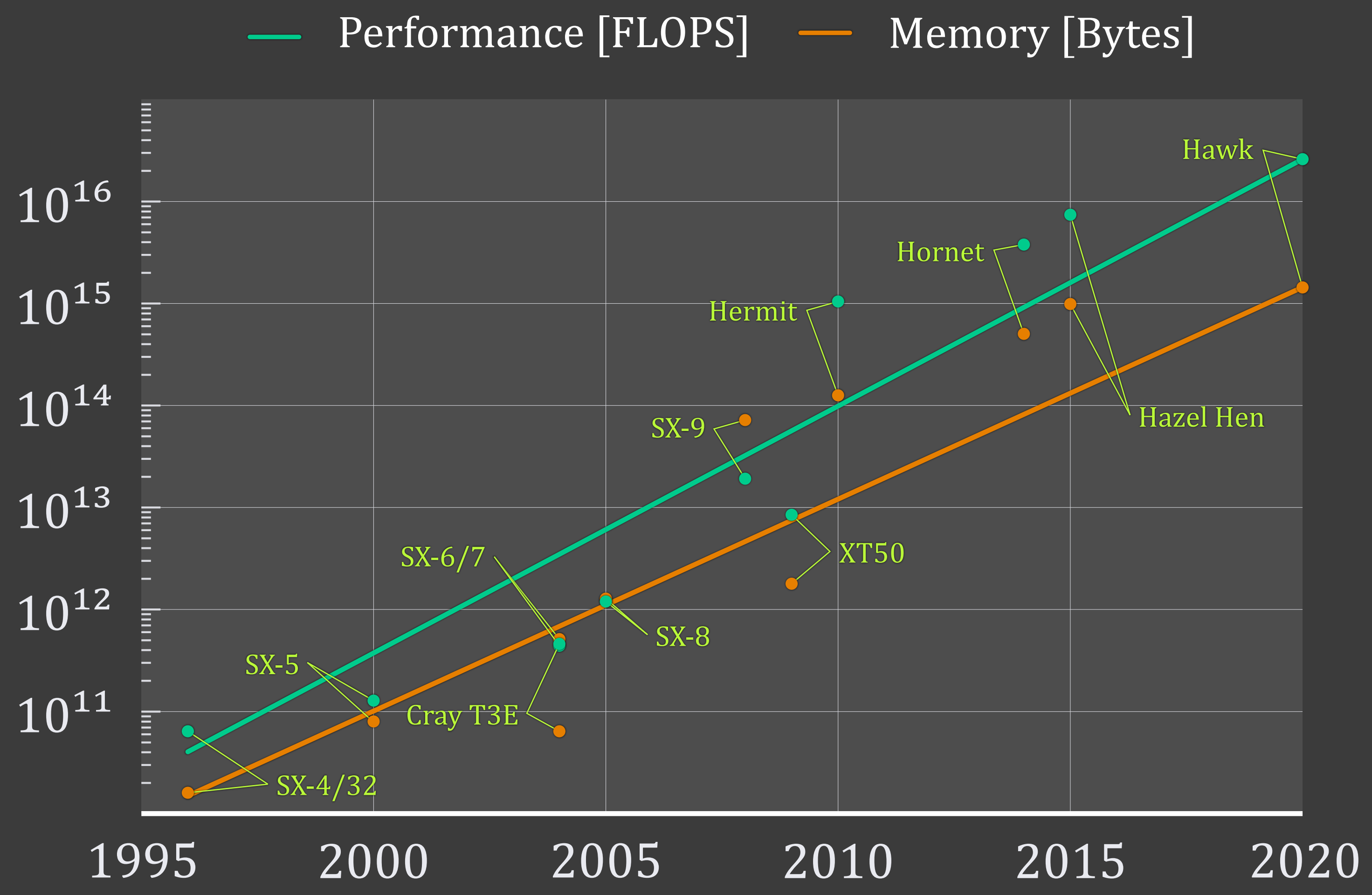
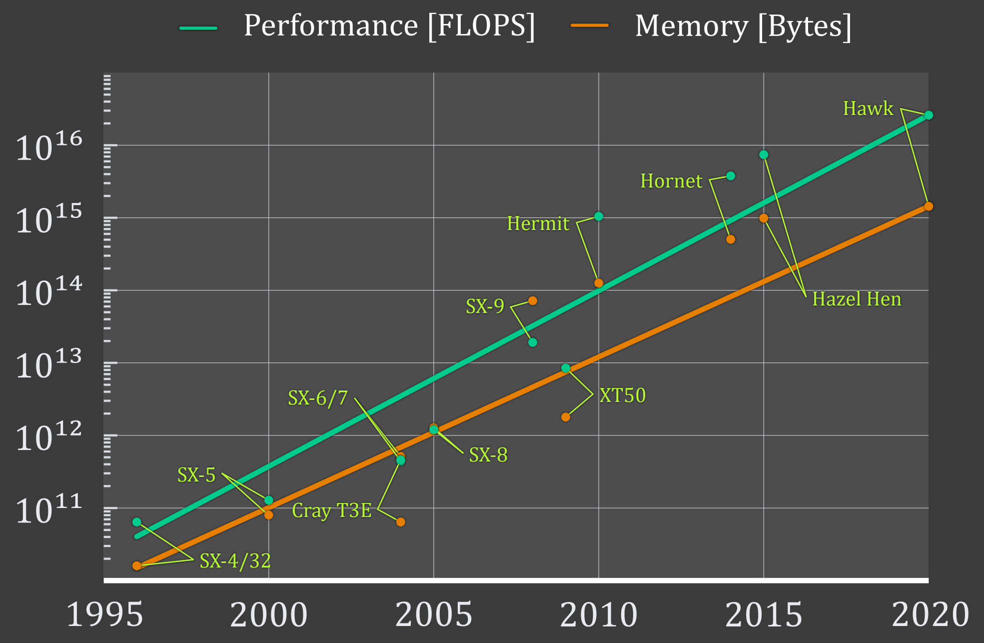


Figure 1 : System memory and performance compared to the launch date for the computer clusters installed at HLRS.

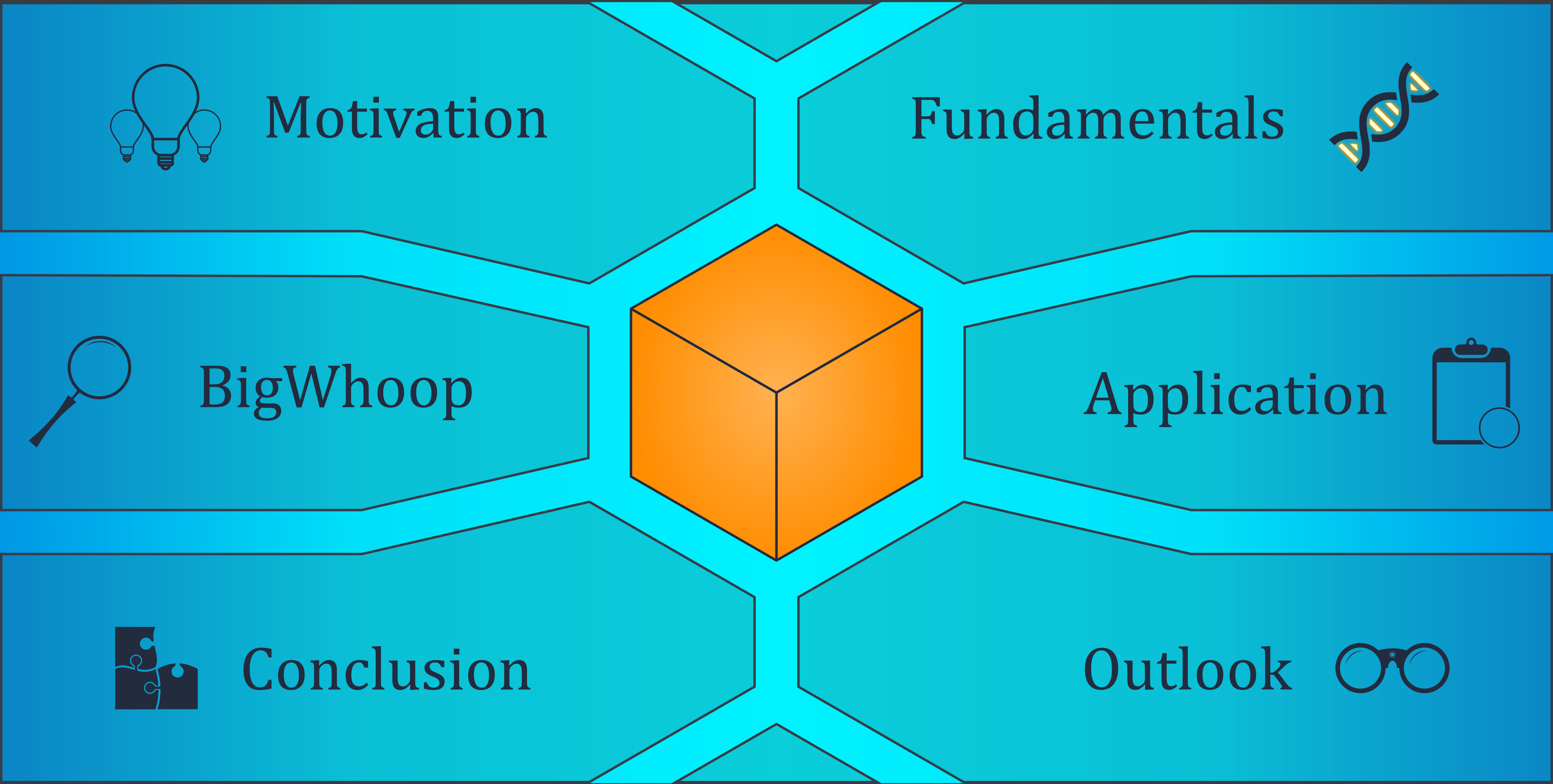
Data Deluge: The Road Ahead

- Research into increasingly complex physical problems and the emergence of deep learning methodologies place high demands on modern HPC systems.
- I/O performance and storage gains have not kept pace with increase in computing power.
- Emerging I/O bottleneck due to growing mismatch between ability to produce and store/analyse data.
- Efficient usage of computing resources is vital when tackling modern problems like weather forecast, climate change, pandemic response, etc.
- I/O bottleneck should be addressed by applying computing resources to more efficiently handle data.



Data Deluge: What can we do?

- Reduce the number of simulations.
- Reduce the number of states stored in memory/written to file system.
- Reduce the number of data-points stored in memory/written to file.
- Reduce the precision of each data-point.
- Apply I/O settings optimised for underlying cluster to optimally use underlying resources.
- Apply established data compression methodologies.


Motivation
Fundamentals
BigWhoop
Application
Conclusion
Outlook

Fundamentals: What is Information?

- "Information is a form of transmission of human experience (knowledge)."
- Commonly associated with a state of disorder/uncertainty in classical thermodynamics and information theory.
- Entropy is used to express the uncertainty in the value of a random variable/the outcome of a random process.
- Orderliness represents a low entropy state.
- Total entropy of a closed system does not decrease.
- Possible size reduction is dependent on data-set entropy.

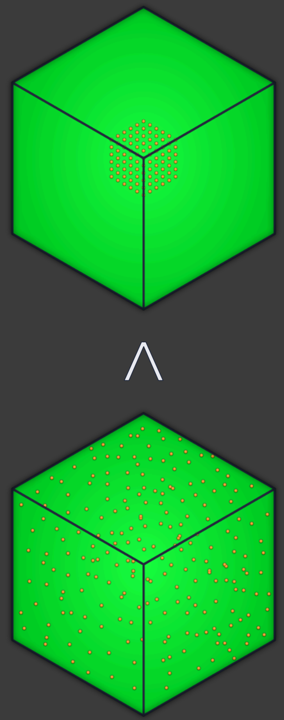


Figure 2 : Example of a low (top) and high (bottom) entropy system.

Fundamentals: What is Compression?

- Encoding of information using as-close-to-optimal number of bits as possible.
- Also known as source coding or bit-rate reduction.
- Exploitation of the inherent statistical redundancies in the information content.
- Subject to time–memory trade-off: higher compression requires more computing resources.
- Available compression techniques include lossless and lossy compression.

Fundamentals: Lossless vs. Lossy Compression?

There shall, in that time, be rumours of things going astray, erm, and there shall be a great confusion as to where things really are, and nobody will really know where lieth those little things wi-- with the sort of raffia work base that has an attachment. At this time, a friend shall lose his friend's hammer and the young shall not know where lieth the things possessed by their fathers that their fathers put there only just the night before, about eight o'clock.

Friends, Romans, countrymen, lend me your ears; I come to bury Caesar, not to praise him. The evil that men do lives after them; The good is oft interred with their bones; So let it be with Caesar. The noble Brutus Hath told you Caesar was ambitious: If it were so, it was a grievous fault, And grievously hath Caesar answer'd it. Here, under leave of Brutus and the rest– For Brutus is an honourable man; So are they all, all honourable men–

Figure 3 : Comparison of information content of the boring prophet speech from The Life of Brian (left) and Marc Anthony's speech in Shakespeare's Julius Caesar (right).

Data Compression: Huffman Coding

- Optimal Prefix Code with variable length for lossless data compression.
- Generated using binary tree (see left).
- Leaf nodes contain symbol and its weight.
- Internal nodes contain links to two children and its weight.
- Symbol with lower weight assigned to left node representing a 0.
- Symbol with higher weight assigned to right node representing a 1.
- Prefix Code for a symbol is read from leaf to root.
- IMPORTANT: No code word represents a prefix for another symbol.

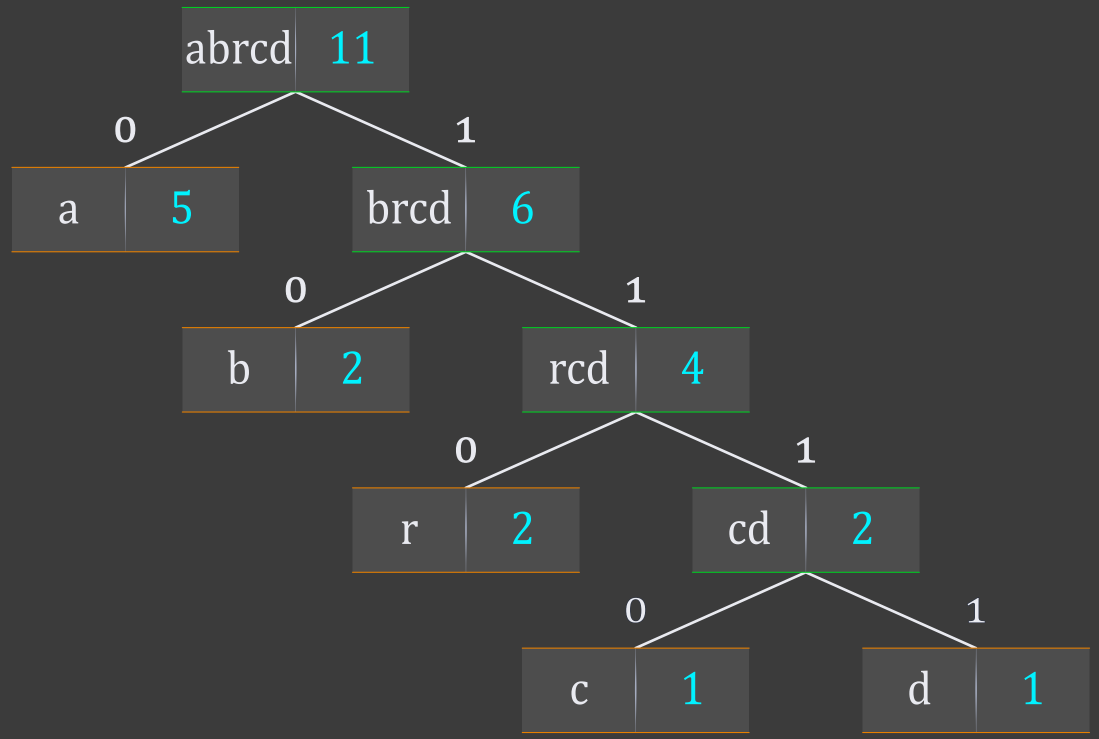


Figure 4 : Huffman binary tree generated from the exact frequencies of the string “abracadabra”

Data Compression: Lossless

- Allows for perfect reconstruction.
- Used in cases were no information loss can be tolerated: binaries, text, documents, source codes etc.
- Forms the foundation for many archive formats (i.e. ZIP) and lossy compression schemes (i.e. JPEG, MP3, H.265).
- Typically involves two steps:
  - Generation of a statistical mode.
  - Mapping of input data points to more efficient bit sequence.

Data Compression: LZ77

- Dictionary encoder using sliding window.
- Search-buffer contains segment of previously seen symbols.
- Look-ahead buffer contains segment of subsequent symbols.
- Encoder generates codeword triples (o,l,c) as binary output:
  - o: Offset between reference and data
  - l: length of the match
  - c: Character following the match
- Forms the basis of many variations including LZW, LZSS, LZMA, etc.

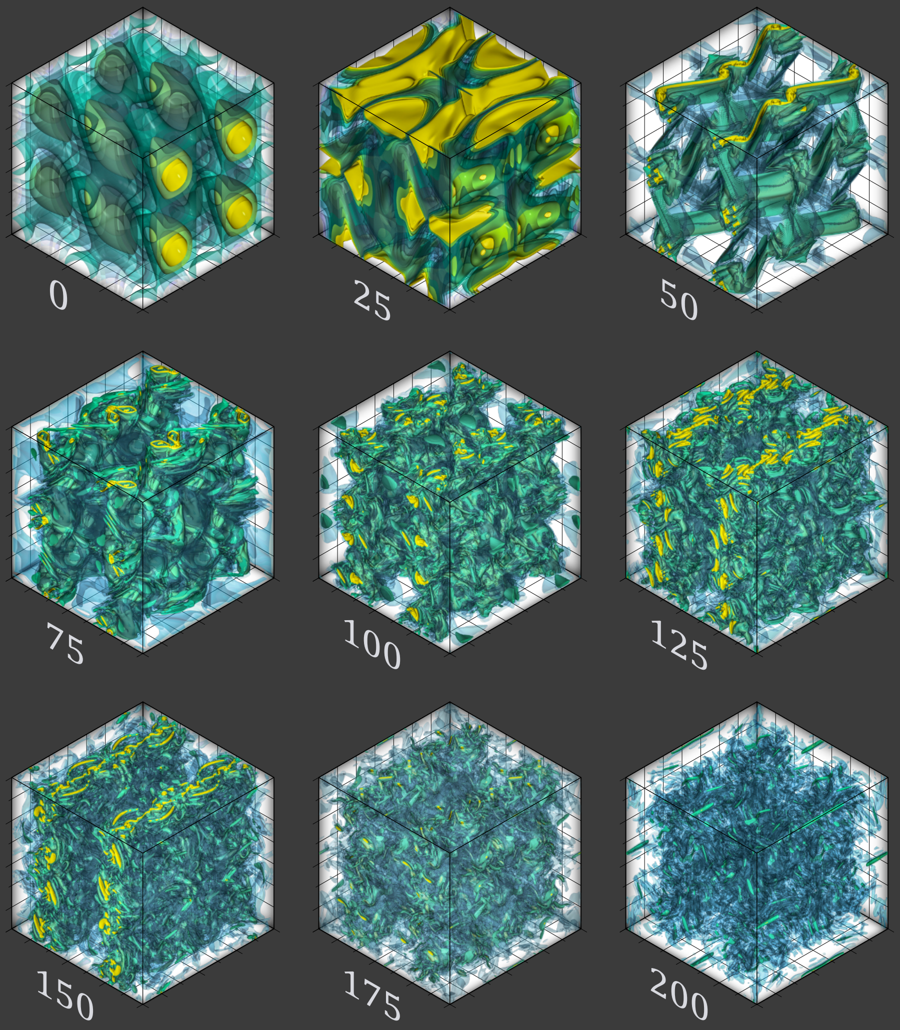
Table 1 : LZ77 Coding example for the string “abracadabra”.

7	6	5	4	3	2	1							
							a	b	r	a	c	ada...	(0,0,a)
						a	b	r	a	c	a	dab...	(0,0,b)
					a	b	r	a	c	a	d	abr...	(0,0,r)
				a	b	r	a	c	a	d	a	bra...	(3,1,c)
		a	b	r	a	c	a	d	a	b	r	a	(2,1,d)
a	b	r	a	c	a	d	a	b	r	a			(7,4,)
a	d	a	b	r	a								

Search-buffer | Look-ahead buffer | Output

Test Case: Taylor Green Vortex

- Simple and well-defined hydrodynamics problem.
- Initial analytical solution containing a single length scale on a 256^3 grid at t = 0.
- Quick transition of initial vortex into fully-turbulent fluid dynamics.
- Broad turbulent scale spectrum useful to study compression performance against different length scales.
- Evaluation performed for the non-dimensional time-steps t =0, 2.5, 5, 7.5, 10, 12.5, 15, 17.5 and 20.
- Size per time-step: 703.052.304 bytes.



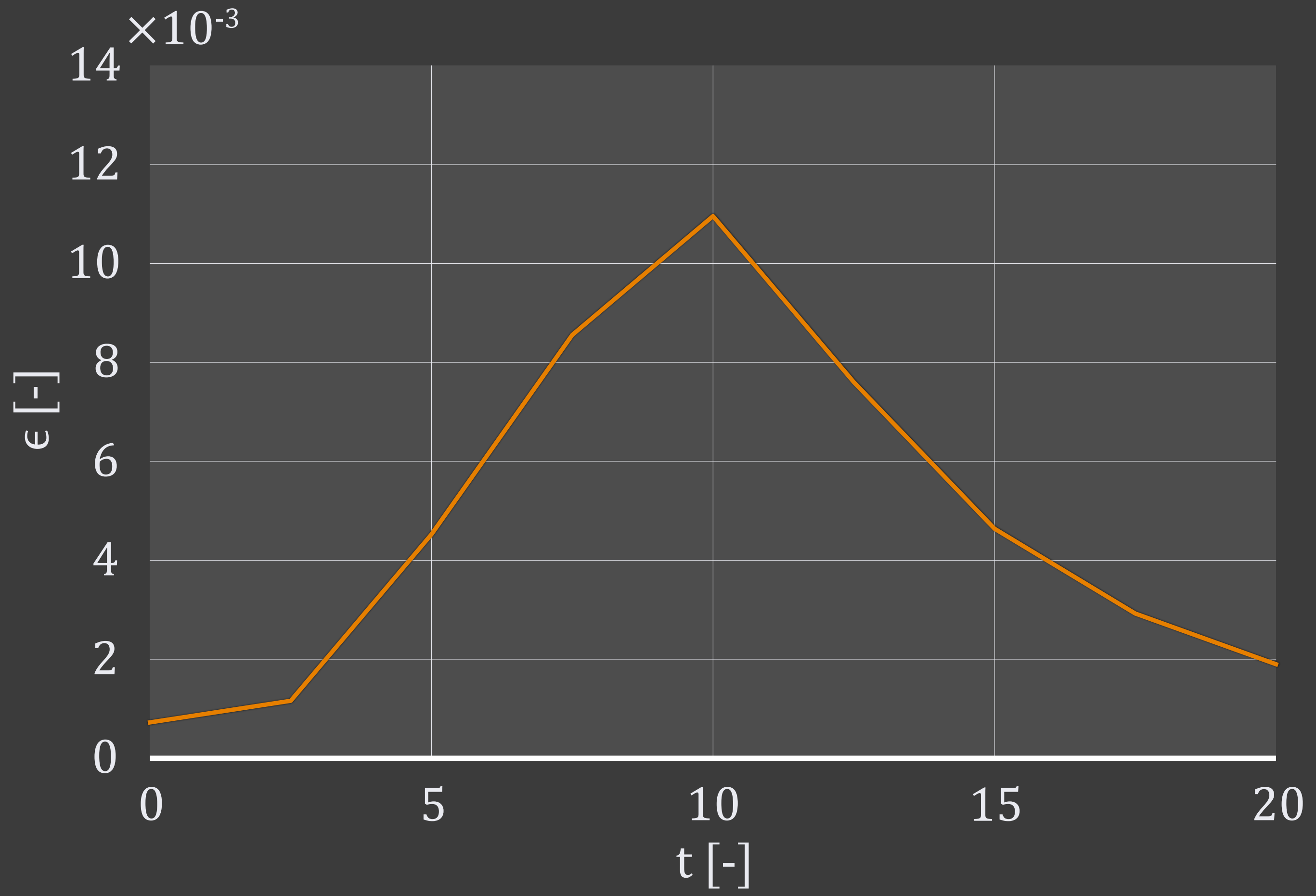


Figure 5 : Normalised energy dissipation rate as a function of the Taylor-Green Vortex evolution over dimensionless time t.

Hands-On: Performance of Dictionary Encoders

Apply the different compressors provided by the 7-Zip command line tool to the Taylor Green Vortex Test Case.

Steps

- Switch to folder: `cd /data`
- Compress a file: `time 7za a -m0=compressor -mx=9 archive.7z filenname`
- Remove archive: `rm archive.7z`

compressor: deflate, lzma, lzma2, bzip2

filename: TGV_0, TGV_50, TGV_100, TGV_150, TGV_200

Metrics

Compression Time

The time needed to process a dataset with a specific compression algorithm. It is typically expressed as the elapsed wall time.

Compression Power

Measurement of the relative reduction in size produced by a data compression algorithm. It is typically expressed as the division of uncompressed size by compressed size.

Peak Signal-to-Noise

Ratio between the maximum possible power of a signal and the power of corrupting noise. PSNR is usually expressed as a logarithmic quantity using the decibel scale.

Results: Performance of Dictionary Encoders

Apply the different compressors provided by the 7-Zip command line tool to the Taylor Green Vortex Test Case.

- Achievable compression performance dependant on information content.
- LZMA2 is fastest.
- LZMA is most efficient.
- Limited compression rate.
- Long compression time.
- Only suitable for data-sets requiring lossless compression.

Table 2 : Compression ratios achieved by the 7-Zip compressors when applied to the Taylor Green Vortex test case.

	0	75	150
Deflate	3.8	1.5	1.5
BZip2	4.8	1.7	1.6
LZMA2	30.1	4.9	3.9
LZMA	34.8	5	4

Data Compression: Suitability of Lossless Compression

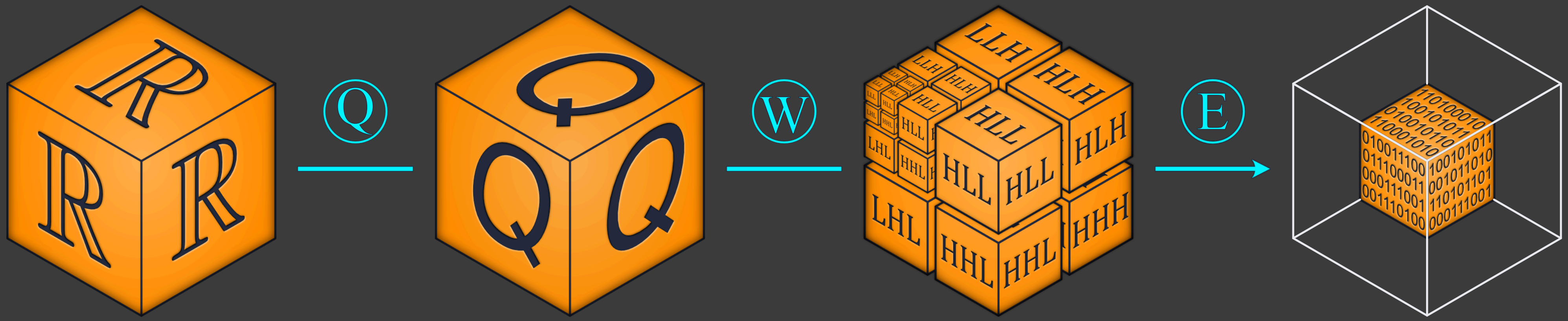
Table 3 : Stuttgart S-Bahn Example. Fields applicable to a lossy compressor appear in turquoise.

Zuggattung	Zugnummer	DS100	Typ	Sollzeit	Prognosezeit	Eingangszeit	Service ID
S	7177	TSC	20	16.09.2017	16.09.2017 00:02:00	15.09.2017 23:58:27	30064857036
S	7177	TSC	20	16.09.2017	16.09.2017	15.09.2017 23:25:13	30064857036
S	7177	TSC	20	16.09.2017	16.09.2017 00:01:00	15.09.2017 23:41:27	30064857036
S	7177	TSC	20	16.09.2017	16.09.2017 00:01:00	15.09.2017 23:33:52	30064857036
S	7177	TSC	20	16.09.2017	16.09.2017	15.09.2017 23:34:53	30064857036
S	7177	TSC	20	16.09.2017	16.09.2017	15.09.2017 23:34:57	30064857036
S	7177	TSC	20	16.09.2017	16.09.2017 00:01:00	15.09.2017 23:22:54	30064857036
S	7177	TSC	20	16.09.2017	16.09.2017	15.09.2017 23:59:24	30064857036
⋮	⋮	⋮	⋮	⋮ ⋮	⋮ ⋮	⋮ ⋮	⋮

Data Compression: Lossy

- Generates an inexact approximation of the original data-set.
- Used in cases were information loss can be tolerated in favour of higher compression ratios: audio, video and image.
- Prominent examples are MP3, H.265 and JPEG.
- Typically involves three steps:
  - Decorrelation
  - Rate Control
  - Entropy Encoding
- Decorrelation typically based on discrete cosine or wavelet transform.

Data Compression: Typical Structure of Lossy Encoder



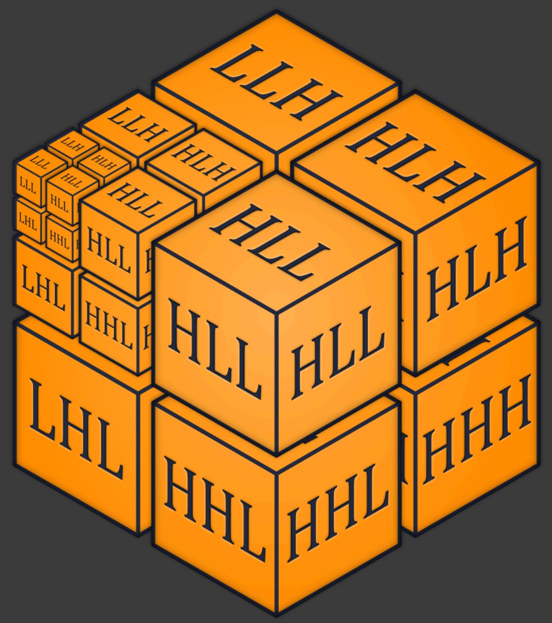
Floating Point Array

Fixed Point Array

Wavelet Coefficients

Compressed Data-stream

Discrete Wavelet Transform



Discrete Cosine Transform

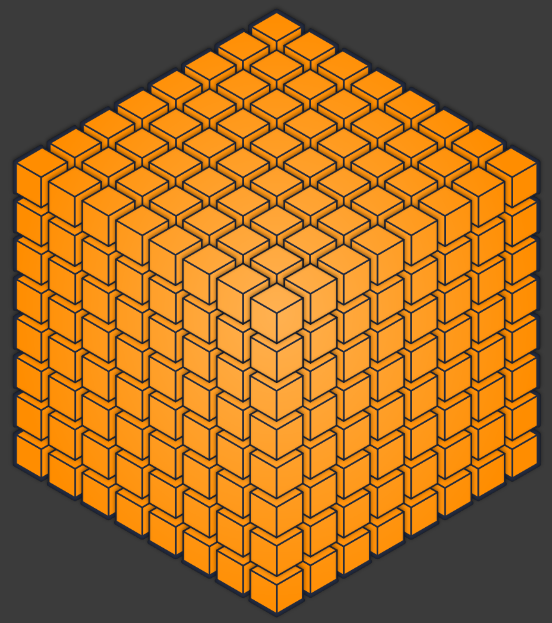


Figure 6 : Symbolic decomposition of a 3-dimensional numerical data set using the discrete wavelet transform (DWT) on the left and discrete cosine transform (DCT) on the right. Application of the wavelet transformation on the entire data set results in sub-bands with loaclized frequency information. Application of the cosine transformation on smaller blocks resulted in local blocks with global frequency information.

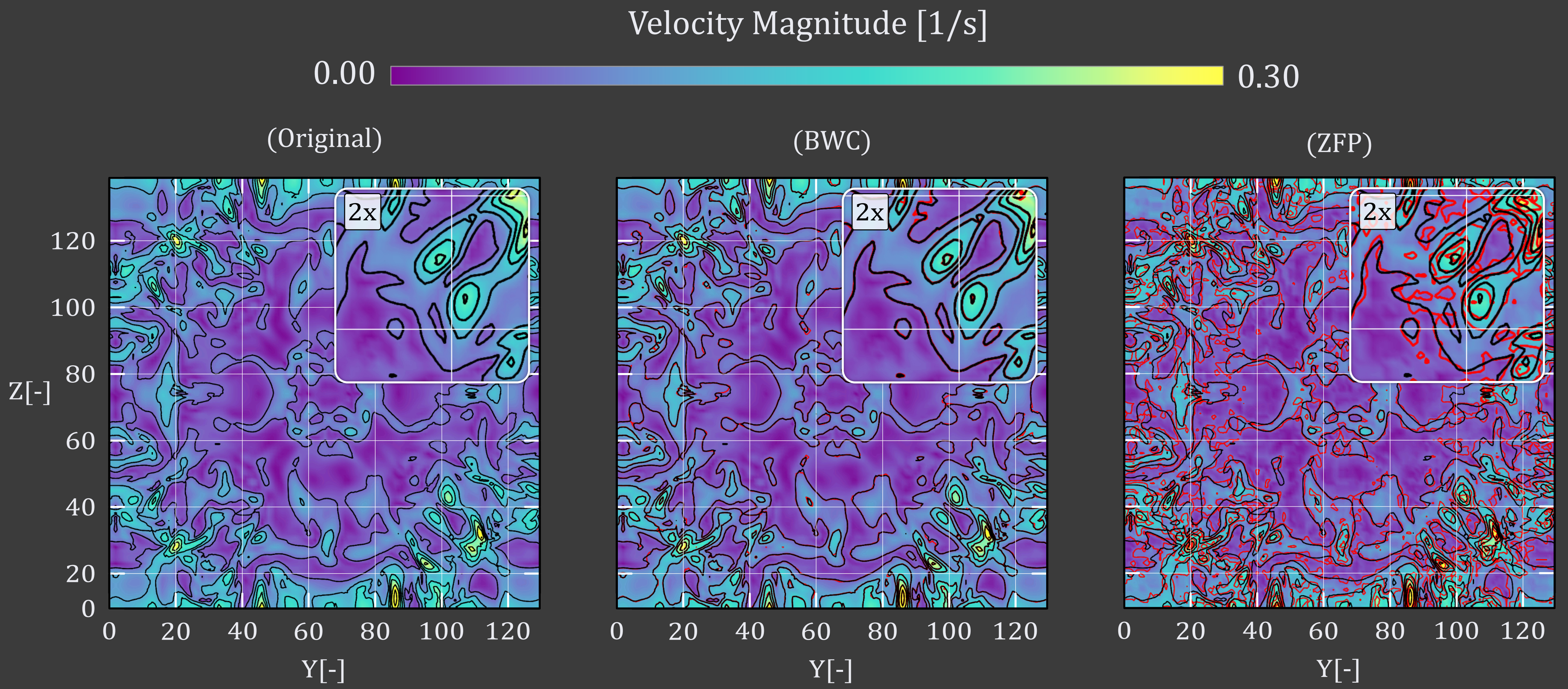


Figure 7 : Contour plots of the vorticity magnitude comparing the original and compressed data sets for the non-dimensional time step $t = 150$ of the TGV test case. Compression was performed with a ratio of 100:1 using the BWC and ZFP libraries. Contour lines are used to highlight the difference between the original (black) and compressed (red) field. Magnification is provided to make the deviation more legible.

Motivation

Fundamentals

BigWhoop

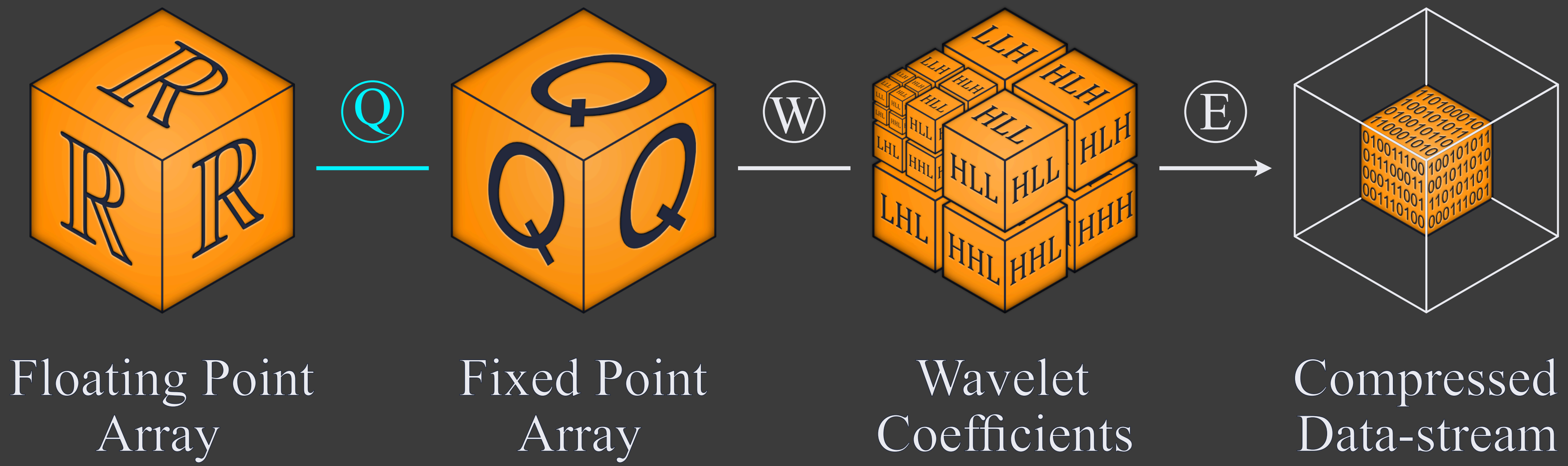
Application

Conclusion

Outlook

BigWhoop Compression Library

- Developed as part of the EU-Projects ExaFLOW and EXCELLERAT.
- Derived from the JPEG 2000 standard for 1- to 4-dimensional IEEE 754 data-sets.
- Designed to facilitate random access as well as region-dependent distortion-control operations.
- Provides an optimally truncated, quality- and resolution scalable code-stream.
- For source code got to: https://code.hlrs.de/TOPIO/BigWhoop
- For questions send an e-mail to: patrick.vogler@hlrs.de

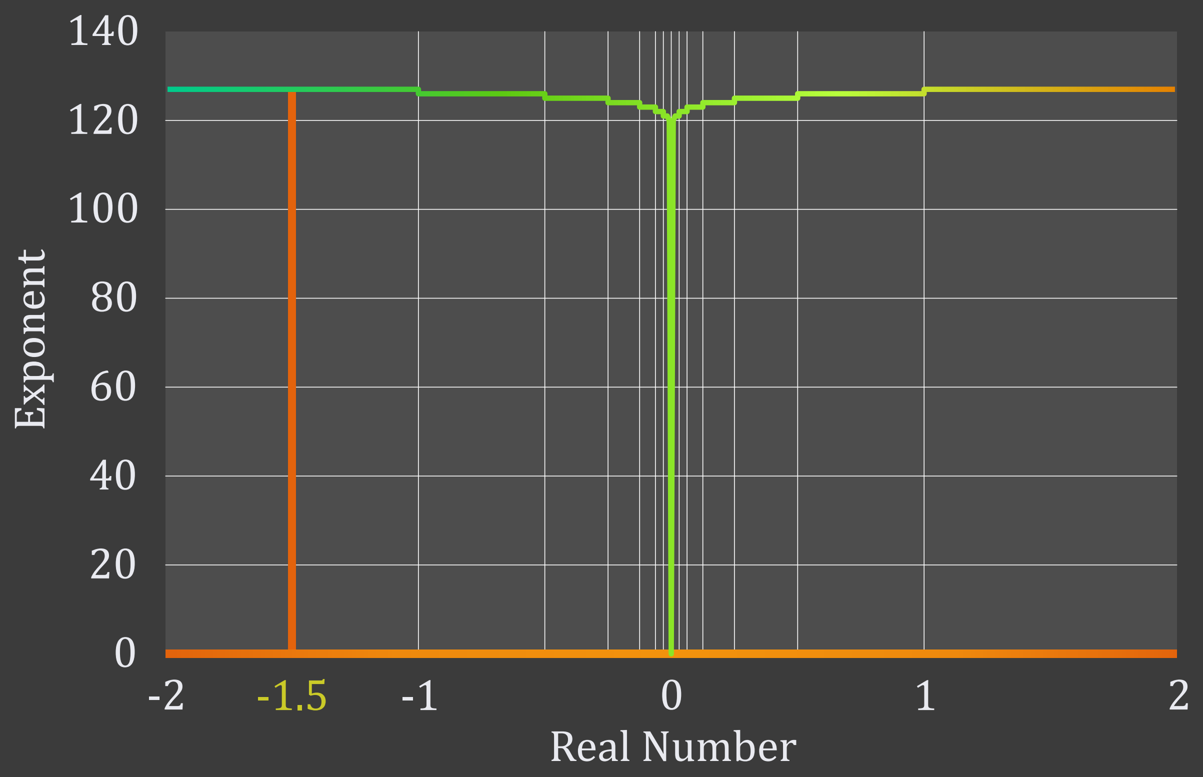

Q
W
E
Floating Point Array
Fixed Point Array
Wavelet Coefficients
Compressed Data-stream

The Real World Problem: Floating-Point Representation

- Floating-point numbers approximate large dynamic range of real numbers.
- Image and video compression standards only defined for integer values.
- IEEE-754 numbers can be split into sign, mantissa and exponent integer fields.
- Non-linearity of mantissa introduces high frequency ‘noise’ on binary level.
- Needlessly reduces compression performance.

$$(-1)^{s} \left(1 + \sum_{i=1}^{23} b_{23-i} \; 2^{-i}\right) 2^{(e-127)}$$

$$-1.5 = (-1)^1 \times (1 + 0.5) \times 2^{(127\ -\ 127)}$$



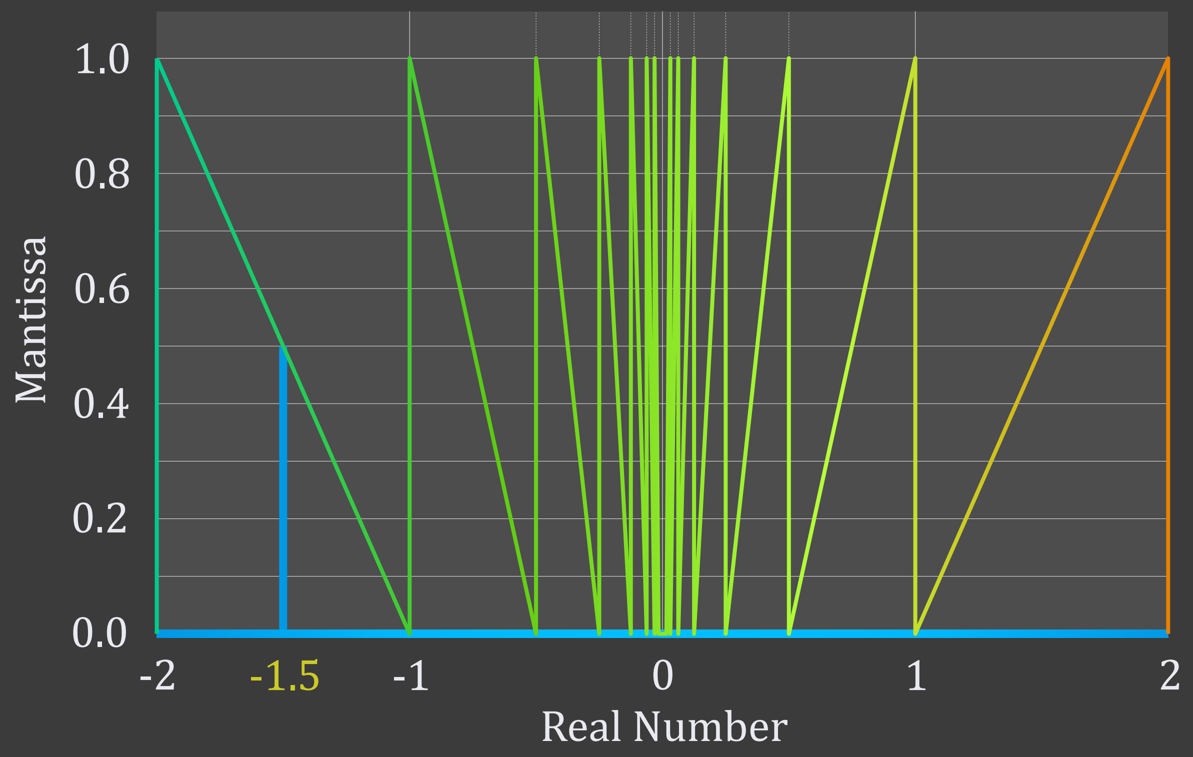


Figure 8 : Progression of the mantissa and biased exponent for the interval $\{x \in R \mid |x| \leq 2\}$ according to the FP32 format. The value -1.5 is highlighted with a sign of 1, mantissa of 0.5 and biased exponent of 127.

The Real World Problem: Fixed-Point Representation

- The number format Q is used to generate a fixed-point representation with constant resolution.
- Dynamic range of each flow-field variable is first centred around zero.
- The values are then normalised to the range (-1, +1).
- The data-points are then multiplied by the number of fractional bits Qm that represent the dynamic range of a synthetic data-type.
- The number of fractional bits should not exceed 62 bits to prevent overflow following the decomposition steps.
- The floating-to-fixed-point transformation can be used for precision reduction prior to compression.

Hands-On: Pre-Compression Precision Reduction

Compress the Taylor Green Vortex test case with the BigWhoop compressor using different number of fractional bits.

Steps

- Compress a file: ../bin/bwc -c filename.eas -qm *1 -n *2 -v 2
- Decompress file: ../bin/bwc -d filename.bwc -n *2 -v 2
- Analyse file: ../bin/bwc -al -i filename_d.eas -rf filename.eas

filename: TGV_0, TGV_50, TGV_100, TGV_150, TGV_200 *1 $=\{x \in \mathbb{Z} \mid 1 \le x \le 61\}$ *2 $=\{x \in \mathbb{Z}\}$

Metrics

Compression Time

The time needed to process a dataset with a specific compression algorithm. It is typically expressed as the elapsed wall time.

Compression Power

Measurement of the relative reduction in size produced by a data compression algorithm. It is typically expressed as the division of uncompressed size by compressed size.

Peak Signal-to-Noise

Ratio between the maximum possible power of a signal and the power of corrupting noise. PSNR is usually expressed as a logarithmic quantity using the decibel scale.

Hands-On: Pre-Compression Precision Reduction

Compress the Taylor Green Vortex test case with the BigWhoop compressor using varying number of fractional bits.

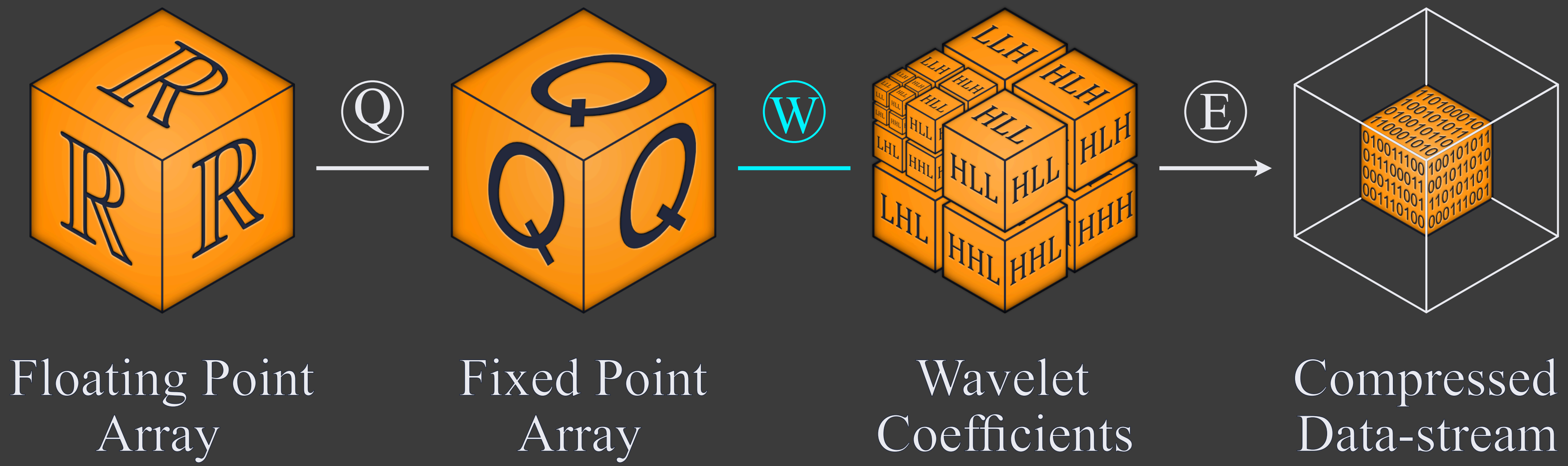
Table 4 : Compression ratio, compression time and peak signal-to-noise ratio (PSNR) for the BigWhoop compressor when applied to the Taylor Green Vortex test with varying Q-number format fractional bits.

Qm	Compression Ratio					Peak Signal-to-Noise Ratio [dB				
	0	50	100	150	200	0	50	100	150	200
1	423263	406617	389280	447226	498963	15.1	13.7	12.8	14.0	15.4
2	227668	188786	191930	178981	195561	21.8	20.6	17.9	16.8	18.4
4	104557	55401	32593	31623	35159	35.2	29.1	23.0	22.2	23.6
8	40681	4102	847	881	1089	55.0	44.6	38.3	37.7	39.3
16	3217	103	24	22	23	94.0	85.6	80.7	80.1	81.1
32	31	6	4	3	3	180.0	177.4	177.0	177.0	178.1
64	3.6	1.6	1.3	1.3	1.3	187.7	188.0	187.8	189.8	190.7

Hands-On: Pre-Compression Precision Reduction

Compress the Taylor Green Vortex test case with the BigWhoop compressor using varying number of fractional bits.

- Pre-compression precision reduction effective tool for performance improvements.
- Compression ratio improvement most notable for low information content.
- Significant jump in Peak Signal-to-Noise ratio between 32 and 16 fractional bits.
- Qm can and should be set to 32 for most use cases.


Q
W
E
Floating Point Array
Fixed Point Array
Wavelet Coefficients
Compressed Data-stream

Decorrelation: Discrete Wavelet Transform

- The discrete wavelet transform is used to gather the majority of the information content in a small number of wavelet coefficients: Gather as much energy in as little data-points as possible.
- The one-dimensional transform splits the data-set into high- and low-frequency bands.
- The high-frequency bands are typically sparsely populated.
- The low-frequency band should by highly correlated.
- Improvement of the information density by subsequent decompositions of the low-frequency bands.
- Can be extended to multidimensional data-sets by applying the filter banks along every available, spatial dimension successively.

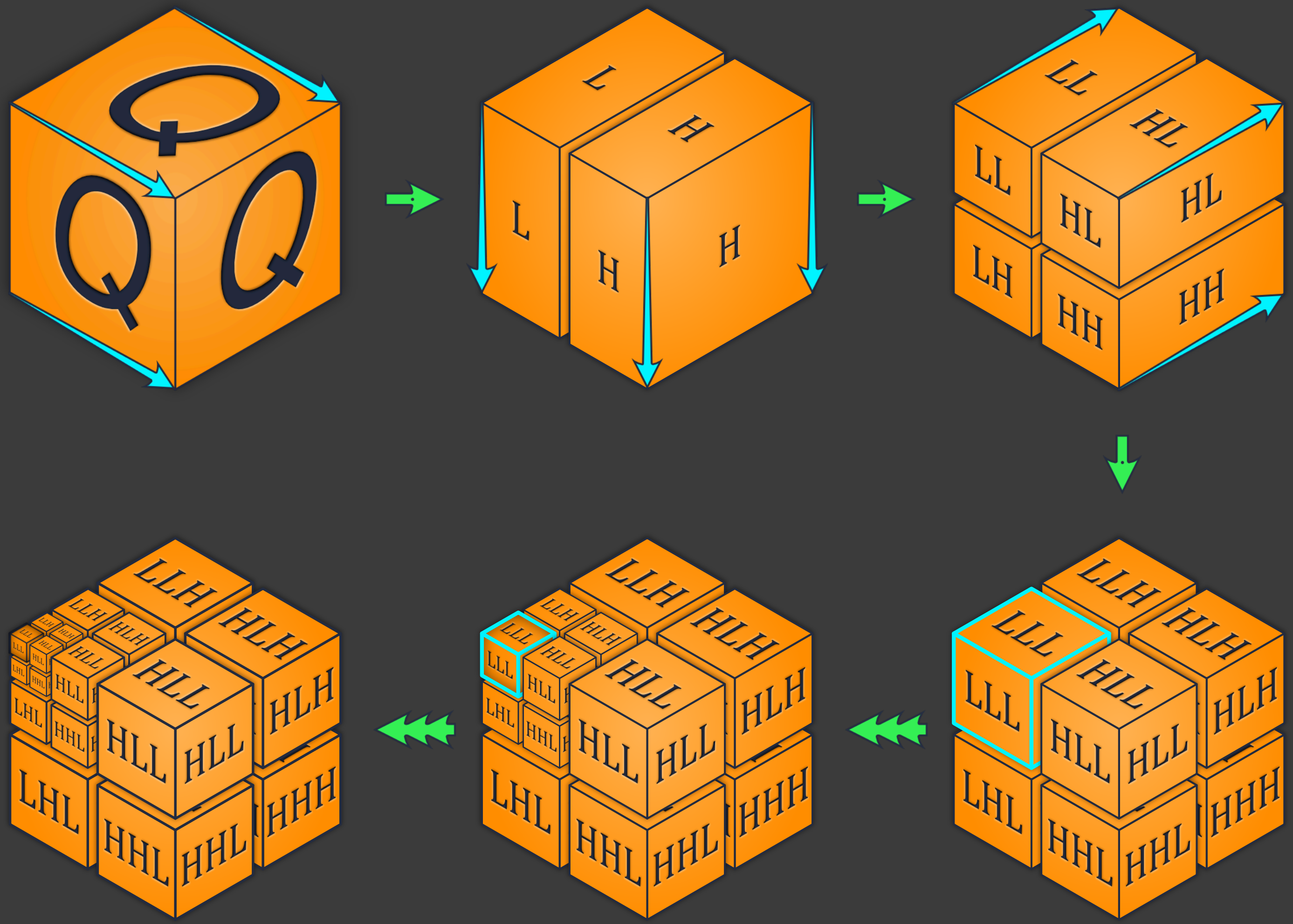


Figure 9 : Symbolic representation of a dyadic wavelet transform applied to a 3-dimensional numerical data set. Turquoise arrows indicate the direction of the one-dimensional wavelet transform, green arrows indicate a single transform step. The position of the letters H and L in a sub-band label implies the directionality of the high and low frequency information.

Hands-On: Effectiveness of Decorrelation

Compress the Taylor Green Vortex test case with the BigWhoop compressor using varying number of decomposition levels.

Steps

- Compress a file: ../bin/bwc -c filename.eas -qm 32 -dl x=*1 y=*1 z=*1 ts=0 -cr 60 -n *2 -v 2
- Decompress file: ../bin/bwc -d filename.bwc -n *2 -v 2
- Analyse file: ../bin/bwc -al -i filename_d.eas -rf filename.eas

filename: TGV_0, TGV_50, TGV_100, TGV_150, TGV_200

*1 $=\{x \in \mathbb{Z} \mid 1 \le x \le 8\}$

*2 $=\{x \in \mathbb{Z}\}$

Metrics

Compression Time

The time needed to process a dataset with a specific compression algorithm. It is typically expressed as the elapsed wall time.

Compression Power

Measurement of the relative reduction in size produced by a data compression algorithm. It is typically expressed as the division of uncompressed size by compressed size.

Peak Signal-to-Noise

Ratio between the maximum possible power of a signal and the power of corrupting noise. PSNR is usually expressed as a logarithmic quantity using the decibel scale.

Hands-On: Effectiveness of Decorrelation

Compress the Taylor Green Vortex test case with the BigWhoop compressor using varying number of decomposition levels.

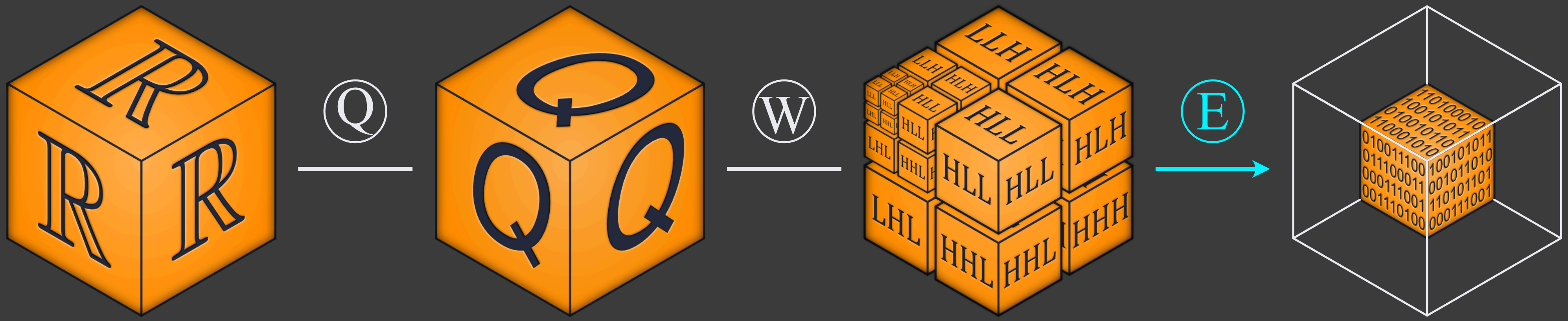
Table 5 : Compression ratio, compression time and peak signal-to-noise ratio (PSNR) for the BigWhoop compressor when applied to the Taylor Green Vortex test with number of decomposition levels.

Level	Compression Ratio					Peak Signal-to-Noise Ratio [dB				
	0	50	100	150	200	0	50	100	150	200
1	60.7	61.7	61.8	61.8	61.8	88.7	68.1	55.4	56.23	58.9
2	62.0	62.6	62.0	62.3	62.1	145.5	91.6	61.6	61.1	64.0
3	61.9	62.6	62.0	62.3	62.0	151.4	89.6	62.2	61.6	64.2
4	62.0	62.6	62.0	62.3	62.0	151.5	95.1	62.3	61.6	64.4
5	62.0	62.6	62.1	62.3	62.0	151.5	95.0	62.2	62.0	64.4
6	62.0	62.6	62.1	62.3	62.1	151.1	95.0	62.2	62.0	64.4
7	63.6	62.6	62.1	62.3	62.1	161.9	95.0	62.2	62.0	64.4
8	85.2	62.7	62.1	62.3	62.1	155.5	77.8	62.2	62.0	54.6

Hands-On: Effectiveness of Decorrelation

Compress the Taylor Green Vortex test case with the BigWhoop compressor using varying number of decomposition levels.

- Wavelet decomposition is necessary to improve compression performance.
- Low frequency information content benefits most from high decomposition level.
- A minimum of 2 decomposition levels is advisable.
- Number of wavelet decomposition ought to be adjusted according to the use-case.
- The Taylor-Green Vortex test case benefits from 7 decorrelation levels.



Floating Point Array

Fixed Point Array

Wavelet Coefficients

Compressed Data-stream

Entropy Encoding: An Overview

- The wavelet coefficients are quantized and stored as an integer array.
- Each sub-band is subdivided into precincts.
- Each precinct is divided into code-blocks
- Each code-block is split into bit-plane fields.
- Every bit-plane is independently processed using the Embedded Block Coding with Optimal Truncation (EBCOT) algorithm.
- The EBCOT algorithm generates symbols and corresponding probabilities that are used for encoding by the arithmetic MQ-Coder.
- The final bit-stream is truncated using rate/distortion estimations to achieve the user defined compression ratio.

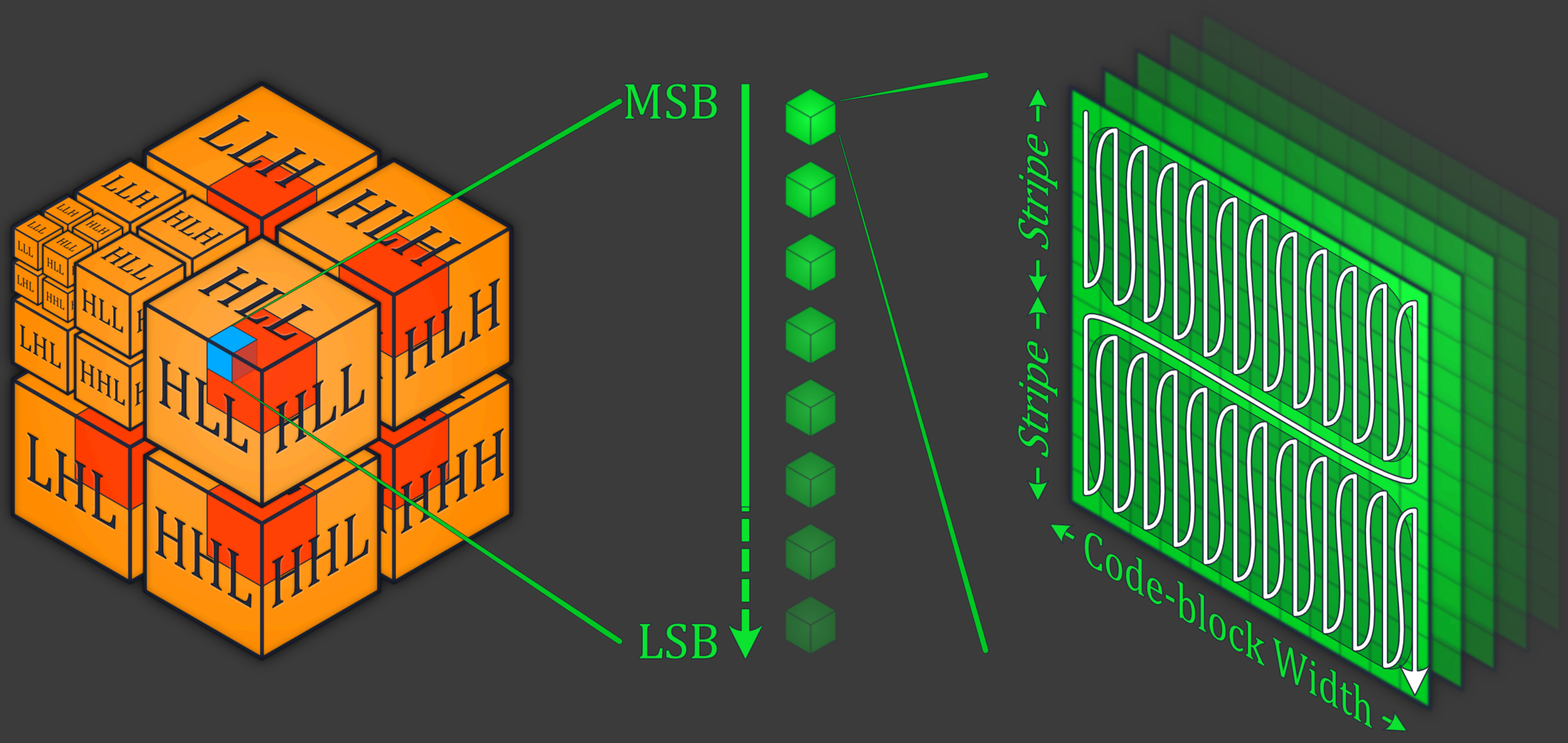
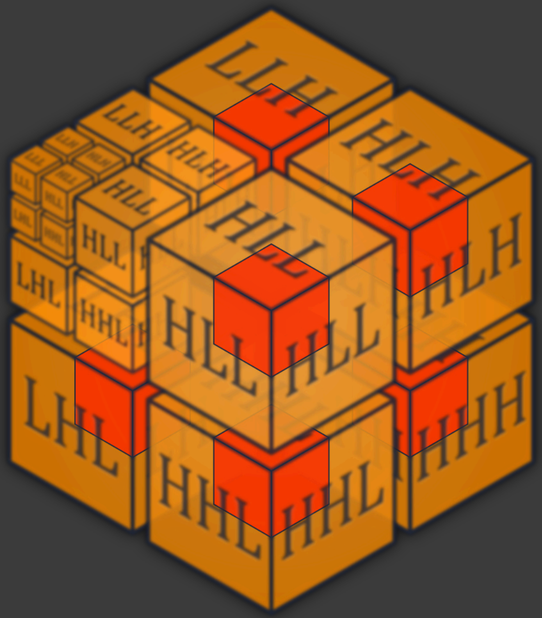


Figure 10 : Geometric operations performed during the **E**mbedded **B**lock **C**oding **O**peration with **O**ptimised **T**runcation (**EBCOT**). Red squares indicate a precinct, green squares a code-block. **MSB** denotes the most significant, **LSB** the least significant bit plane and Stripe the encoding width within a code-block plane.

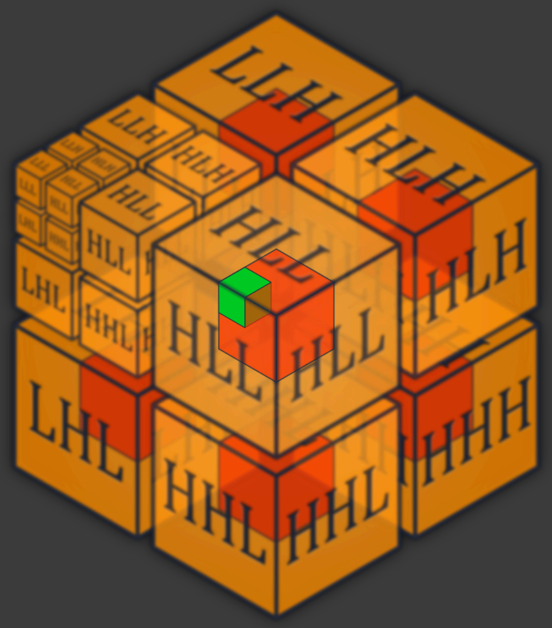
Entropy Encoding: Precincts

- Precinct are defined as non-overlapping, spatially related sub-band regions.
- Represent the upper bound for the code-block size.
- Used for code-stream construction.



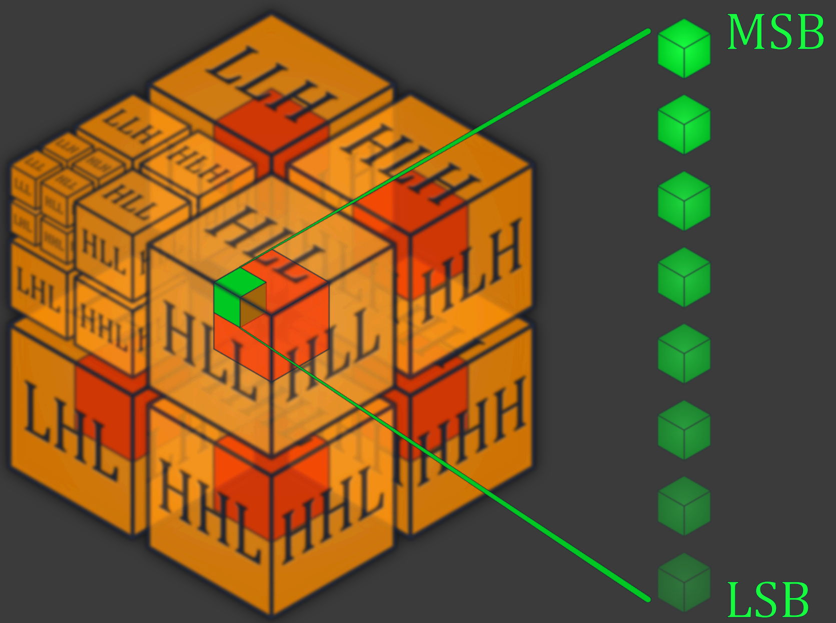
Entropy Encoding: Code-Blocks

- Smallest spatial granulation for code-stream rate control.
- Form the input for the entropy encoding stage.
- Larger code-blocks lead to better compression.
- Smaller code-blocks allow for better rate control and fault tolerance.



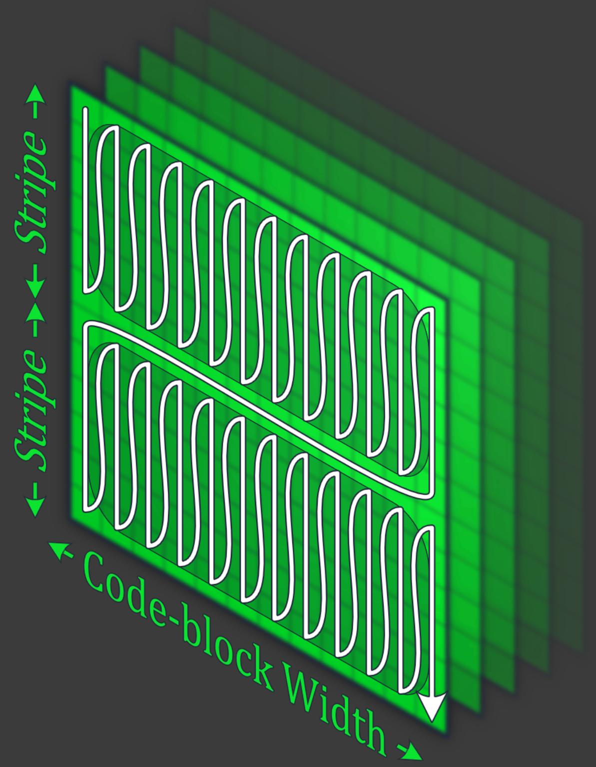
Entropy Encoding: Bit-Planes

- Set of bits corresponding to the same bit position in the binary representation of a data-type.
- The number of bit-planes correspond to the number of fractional bits Qm.
- Valuable information that is most likely retained will be in the vicinity of the Most-Significant-Bit (MSB) plane.
- Insignificant information that is most likely omitted from the code-stream will be close to the Least-Significant-Bit (LSB) plane.
- Coding is carried out from MSB to LSB.



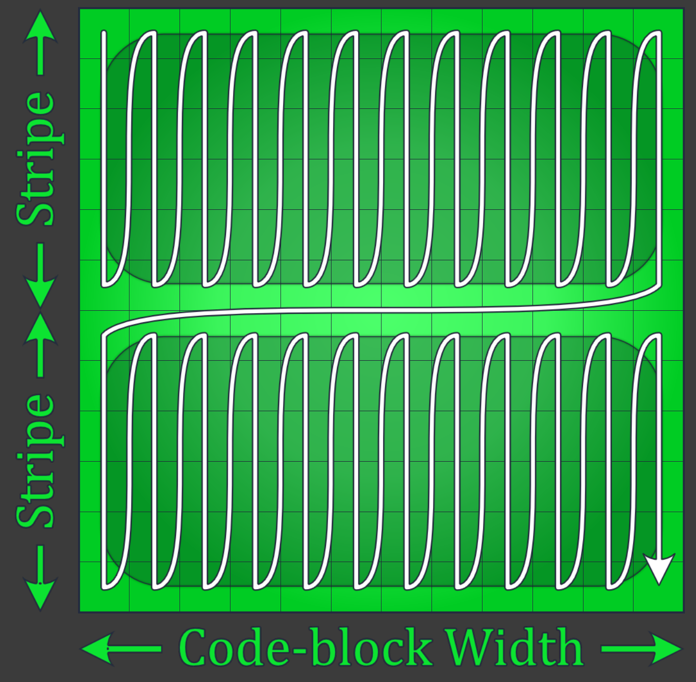
Entropy Encoding: Coding Plane

- Each bit plane is split into 2-dimensional coding planes.
- Each coding plane is scanned independently in a zig-zag pattern.
- Every bit of a bit-plane is only encoded once in one of three coding passes.
- Each coding pass generates a compress data-packet.
- The error introduced by omission of each packet is calculated on the fly.
- The pre-defined compression ratio is achieved by omitting coding passes that introduce the least amount of errors.



Entropy Encoding: Coding Passes

- Significance Propagation
  - Coefficients with significant (non-zero) neighbourhood.
  - Encodes the significance and sign information.
- Refinement
  - Coefficients that have become significant in a previous pass.
  - Encodes the magnitude information.
- Cleanup
  - Coefficients not processed in the previous two passes.
  - First pass to be applied to the code-block.
  - Encodes the significance and sign information.



Hands-On: Explore

Explore the Taylor Green Vortex test case with the BigWhoop compressor using different settings.

Steps

- Compress a file: ../bin/bwc -c filename.eas
  - -qm *1
  - -dl x=*2 y=*2 z=*2 ts=0
  - -cb x=*3 y=*3 z=*3 ts=0
  - -p x=*4 y=*4 z=*4 ts=0
  - -cr *5
  - -n *5
  - -v 2
- Decompress file: ../bin/bwc -d filename.bwc -n *5 -v 2
- Analyse file: ../bin/bwc -al -i filename_d.eas -rf filename.eas

filename: TGV_0, TGV_50, TGV_100, TGV_150, TGV_200

*1 $=\{x \in \mathbb{Z} \mid 1 \leq x \leq 61\}$

*2 $=\{x \in \mathbb{Z} \mid 1 \leq x \leq 8\}$

*3 $=\{x \in \mathbb{Z} \mid 3 \leq x \leq 6\}$

*4 $=\{x \in \mathbb{Z} \mid 3 \leq x \leq 8\}$

*5 $=\{x \in \mathbb{Z}\}$

Metrics

Compression Time

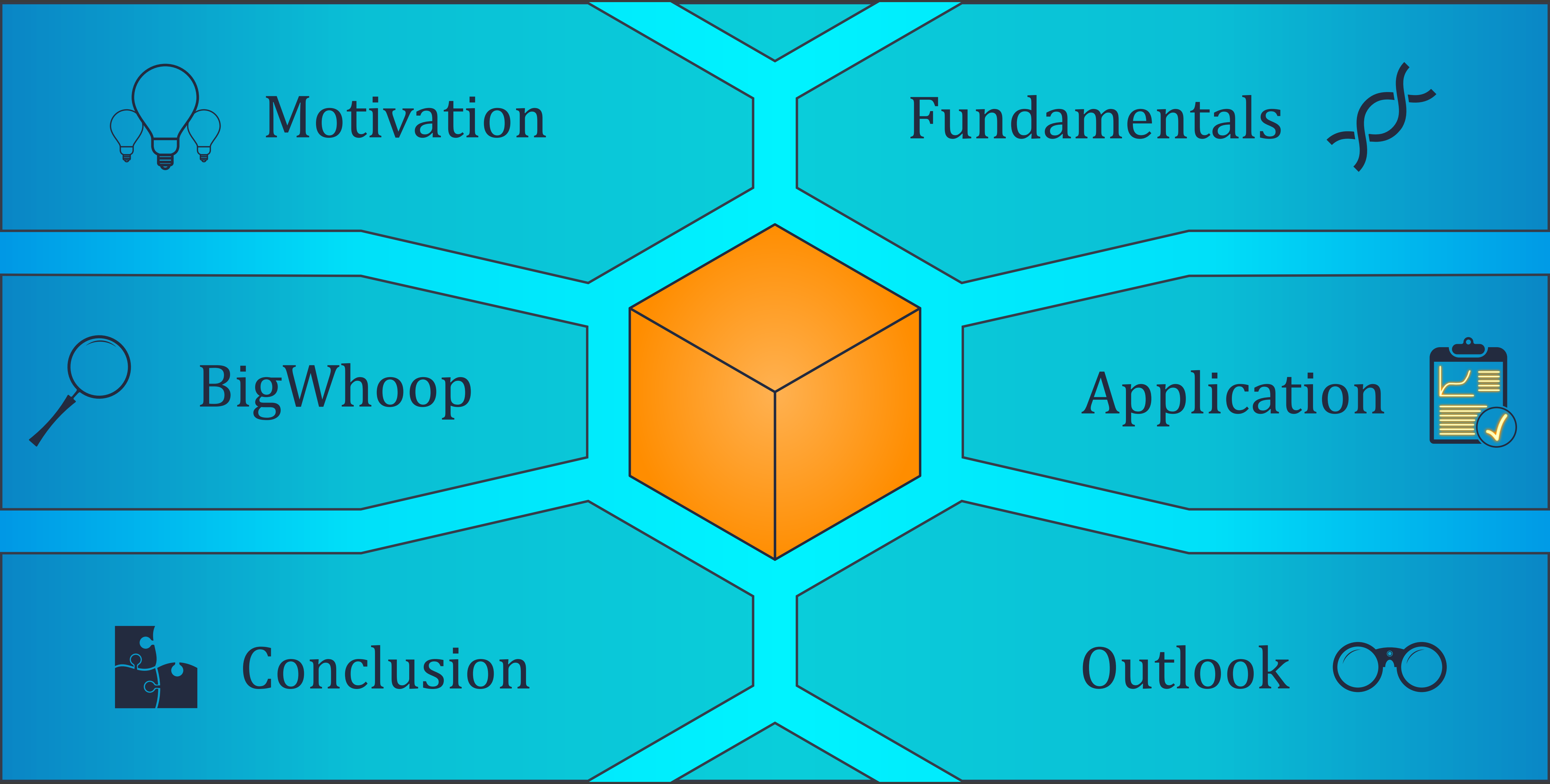
The time needed to process a dataset with a specific compression algorithm. It is typically expressed as the elapsed wall time.

Compression Power

Measurement of the relative reduction in size produced by a data compression algorithm. It is typically expressed as the division of uncompressed size by compressed size.

Peak Signal-to-Noise

Ratio between the maximum possible power of a signal and the power of corrupting noise. PSNR is usually expressed as a logarithmic quantity using the decibel scale.


Motivation
Fundamentals
BigWhoop
Application
Conclusion
Outlook

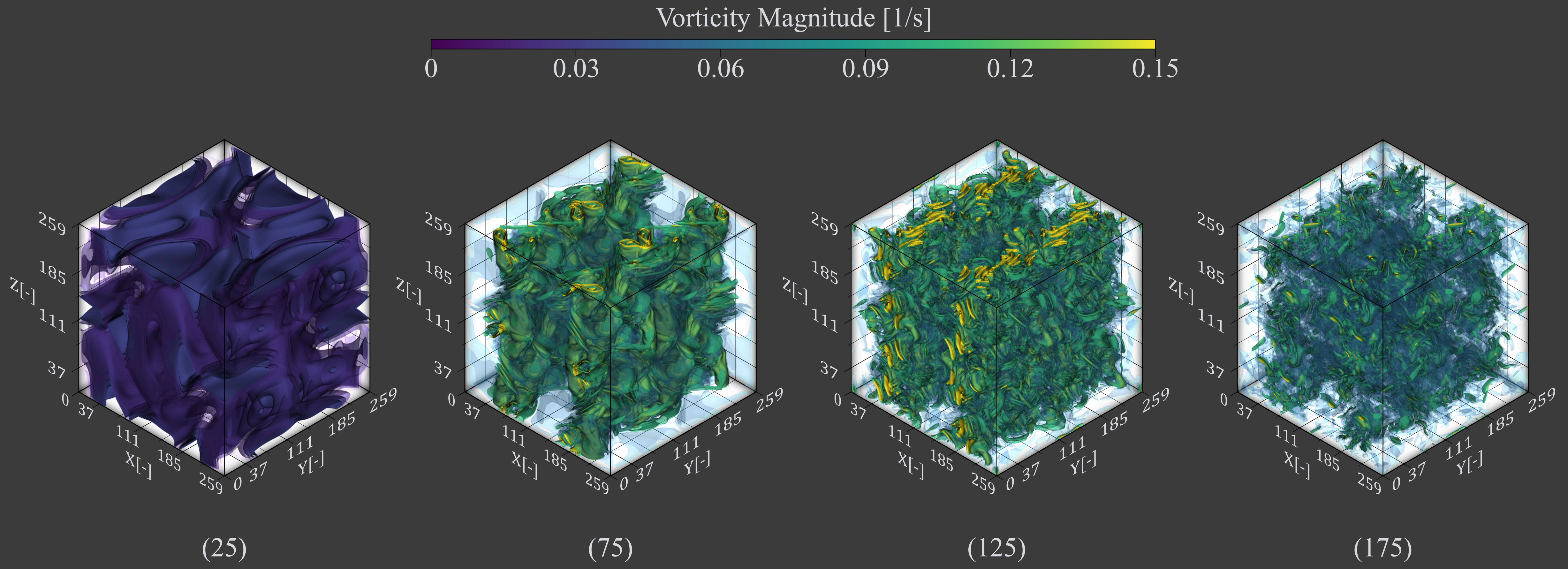


Figure 11 : Visualisation of the Taylor-Green Vortex test case on a 256^3 grid. The vorticity magnitude is shown from left to right for the non-dimensional times t= 25, 75, 125 and 175.

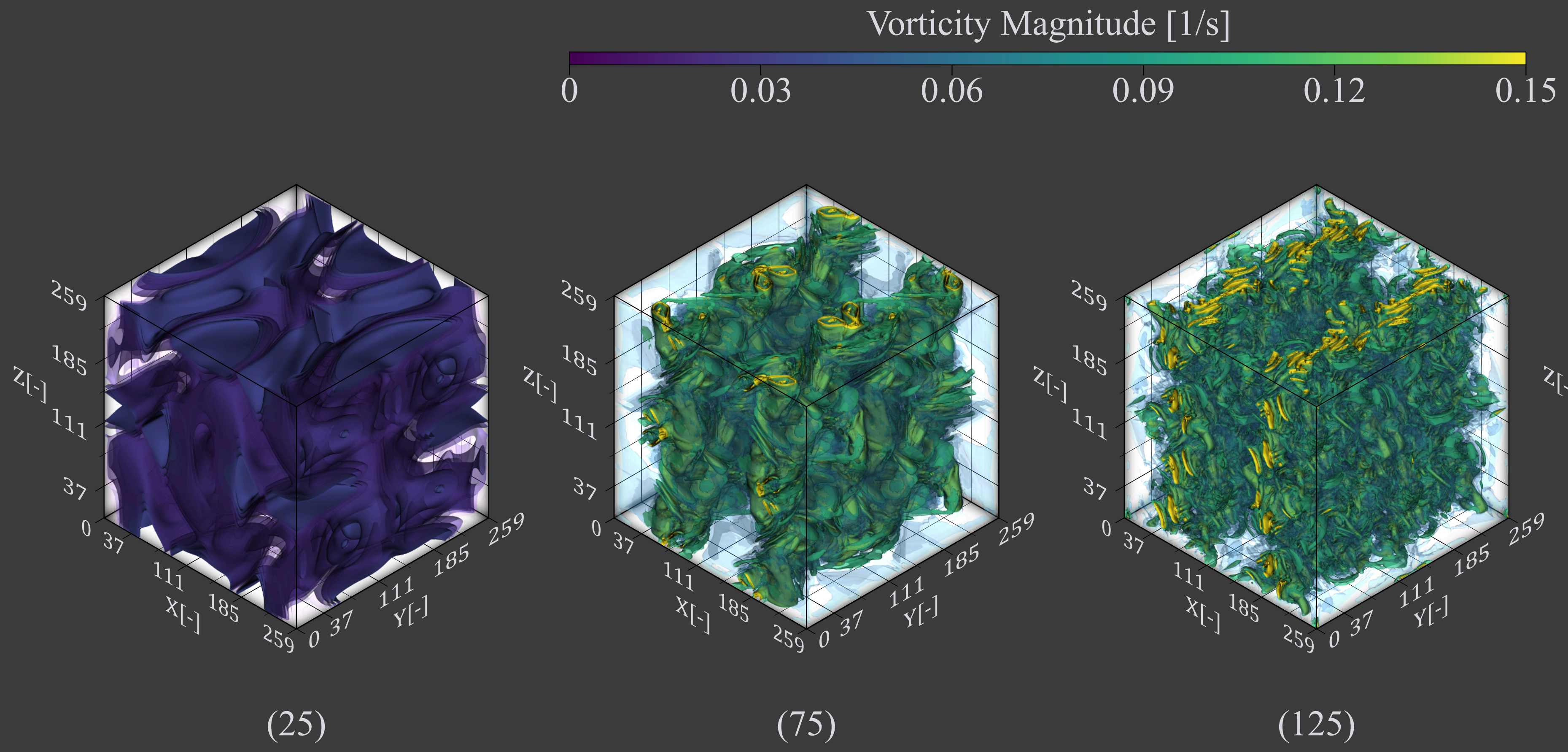


Figure 12 : Visualisation of the decompressed Taylor-Green Vortex test case on a 256^3 grid for a compression ratio of 1 : 400. The vorticity magnitude is shown from left to right for the non-dimensional times t= 25, 75, 125 and 175.

Table 6 : Compression ratio, compression time and peak signal-to-noise ratio (PSNR) for the BigWhoop, ZFP and 7-Zip compressor. The- best results appear in turquoise.

Time-step	Compression Ratio			Compression Time			Peak Signal-to-Noise Ratio		
	BWC	ZFP	7-Zip	BWC	ZFP	7-Zip	BWC	ZFP	7-Zip[a]
0	64.7	65.0	41.7	6.9	1.7	114.9	133.9	61.2	∞
2.5	62.2	62.1	6.0	4.4	2.0	160.4	123.6	60.0	∞
5	61.6	61.1	5.8	7.2	2.0	173.2	92.0	48.2	∞
7.5	61.0	61.1	5.6	8.8	1.9	164.0	73.8	44.3	∞
10	60.8	61.1	5.1	11.0	1.8	172.9	62.6	37.9	∞
12.5	60.8	61.1	4.6	11.4	1.8	172.8	63.3	40.6	∞
15	60.8	61.1	4.2	11.1	1.8	186.4	64.1	40.5	∞
17.5	60.8	61.1	3.9	10.7	1.8	183.9	65.7	43.0	∞
20	60.9	61.1	3.7	11.1	1.9	181.5	67.1	45.2	∞

[a] The PSNR for the 7- Zip compressor is always infinity due to its lossless nature

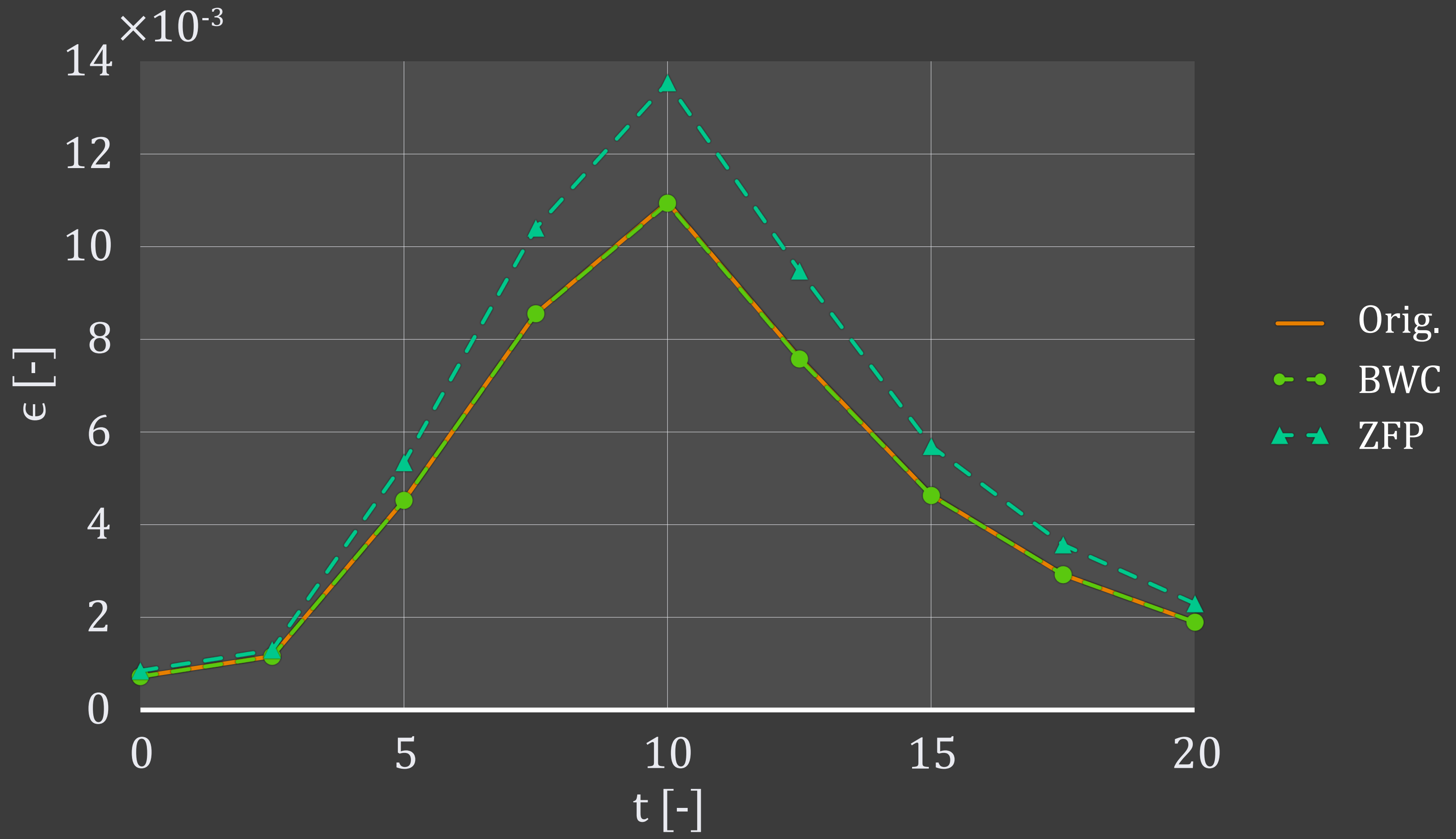


Figure 13 : Comparison of normalised energy dissipation rate as a function of the Taylor-Green vortex evolution at a compression ratio of 120:1 for the non-dimensional time-steps t = 0, 25, 50, 75, 100, 125, 150, 175, 200.

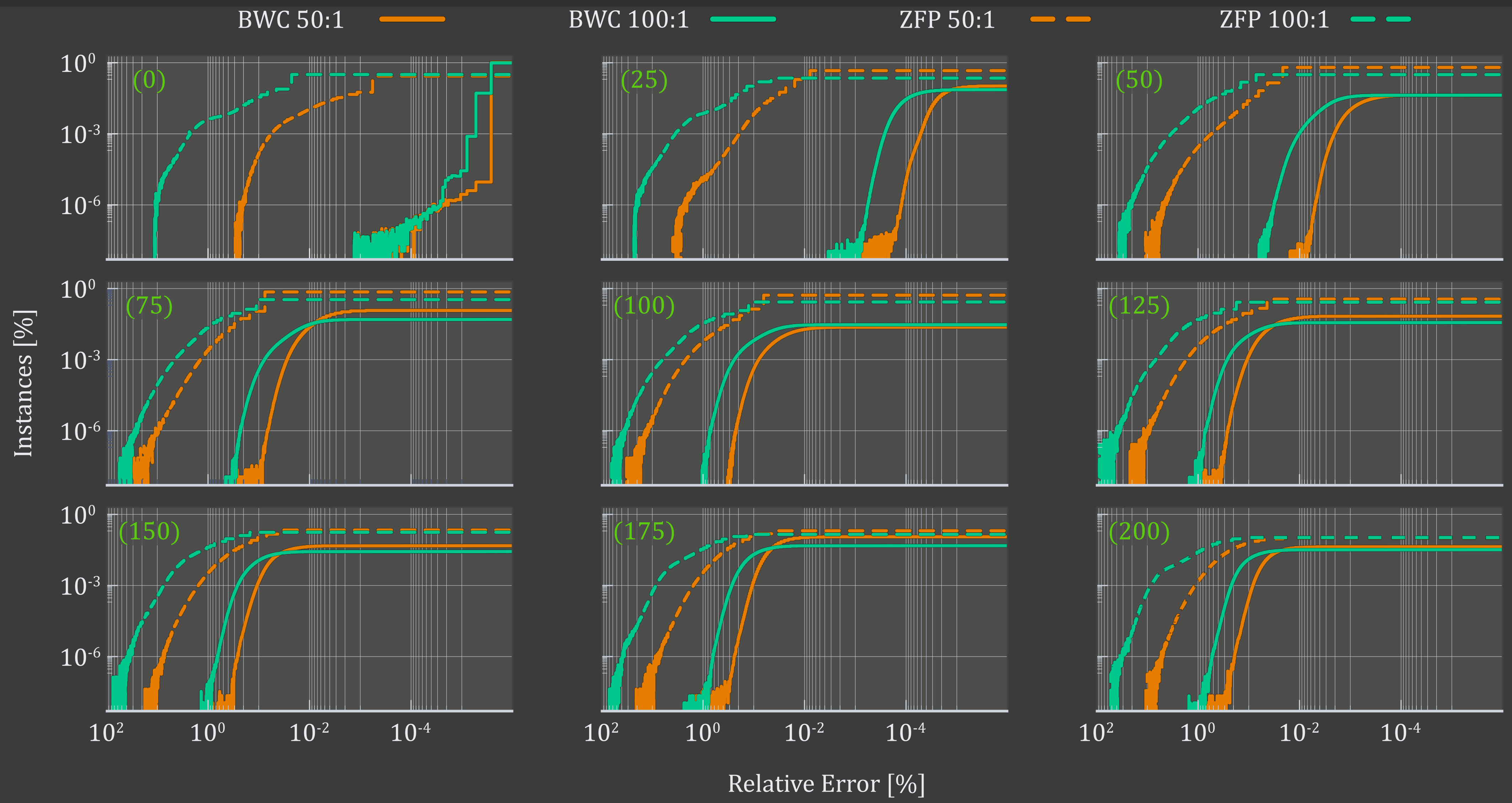
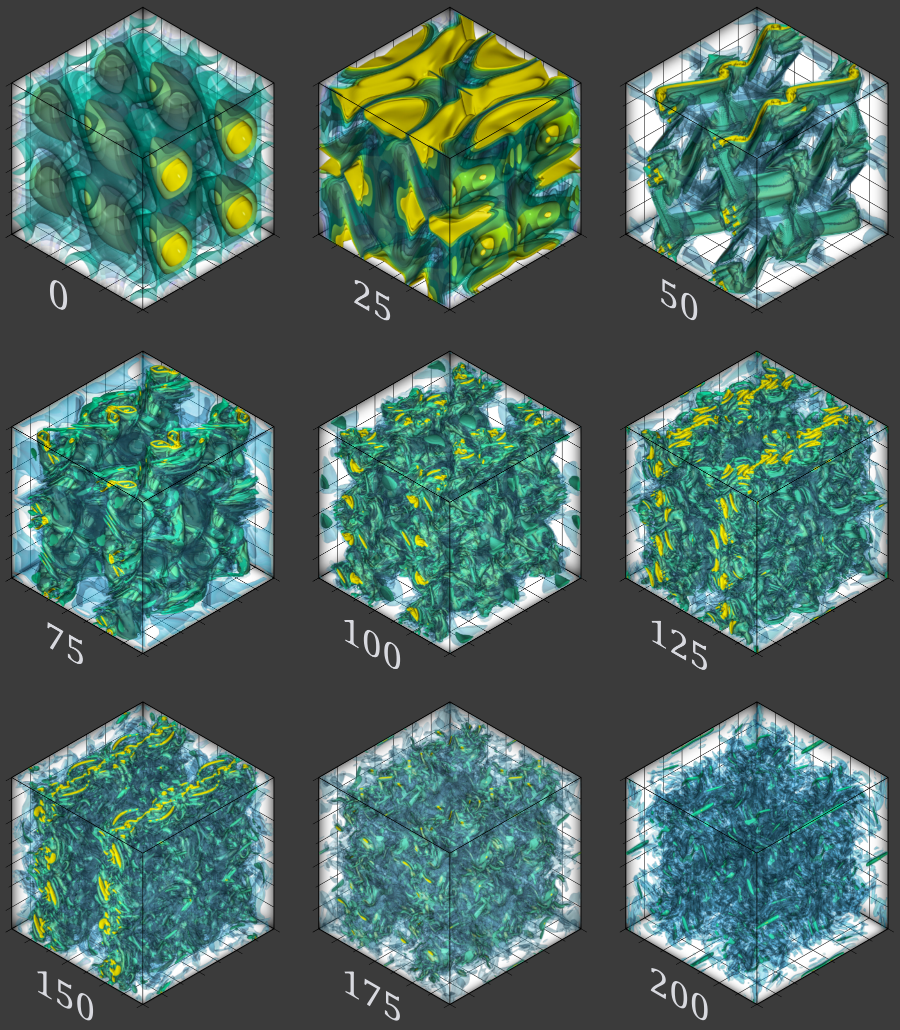


Figure 14 : Histogram plot of the relative error for the non-dimensional time steps ($t = 0 \dots 200$) of the Taylor-Green vortex test case. The original data set is compared with BWC and ZFP compressed data sets with a compression ratio of 50:1 and 100:1. The evaluation was performed for the variables ρ, ρE, ρu0, ρu1 and ρu2 on 500 bins of equal width.

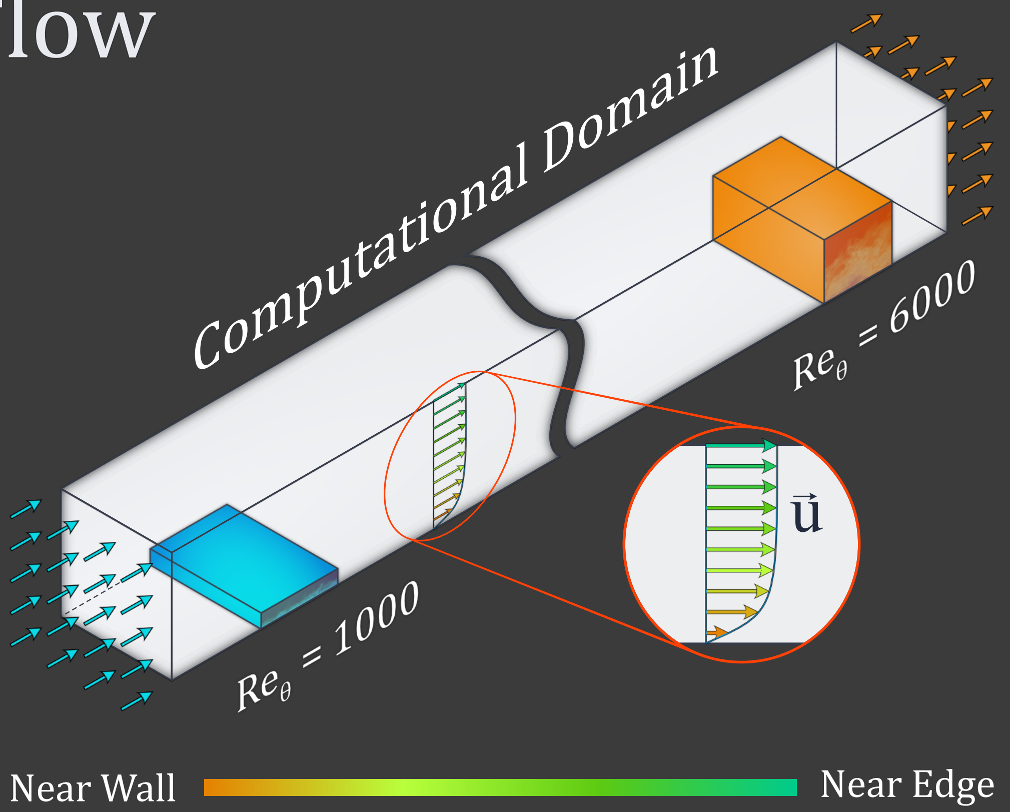
Taylor Green Vortex: Results

- Dictionary encoders are slow, offer modest compression but deliver perfect reconstruction.
- 7zip is not suitable for large numerical data-sets.
- Block transform (ZFP) based compressor is fast, offers good compression and fair reconstruction.
- ZFP is unsuitable for post-processing of turbulence statistics due to its tendencies to dissipate a lot of the high-freq. information.
- Discrete Wavelet Transform (BWC) based compression is reasonably fast, offers good compression and excellent reconstruction.
- BigWhoop can be used for demanding pre-processing applications requiring good information retention.



Test Case: Turbulent Flat Plate Flow

- Direct numerical simulation of a turbulent flow over a flat plate at Ma = 2.0.
- Reynolds-Number range of $Re\theta = 300$ - 6600 on a $nx \times ny \times nz = 17200 \times 620 \times 1920$ grid.
- Size per time-step: 818.995.200.000 bytes.
- Encoding of entire simulation and sub-domain at compression ratios 1 : 50, 1 : 100 and 1 : 200.
- Compression performance evaluated for sub-domains at in- and outlet of the complete domain as well as the sub-domain at the outlet.
- Error Histogram calculated on wall-parallel, 10 grid points thick slices.



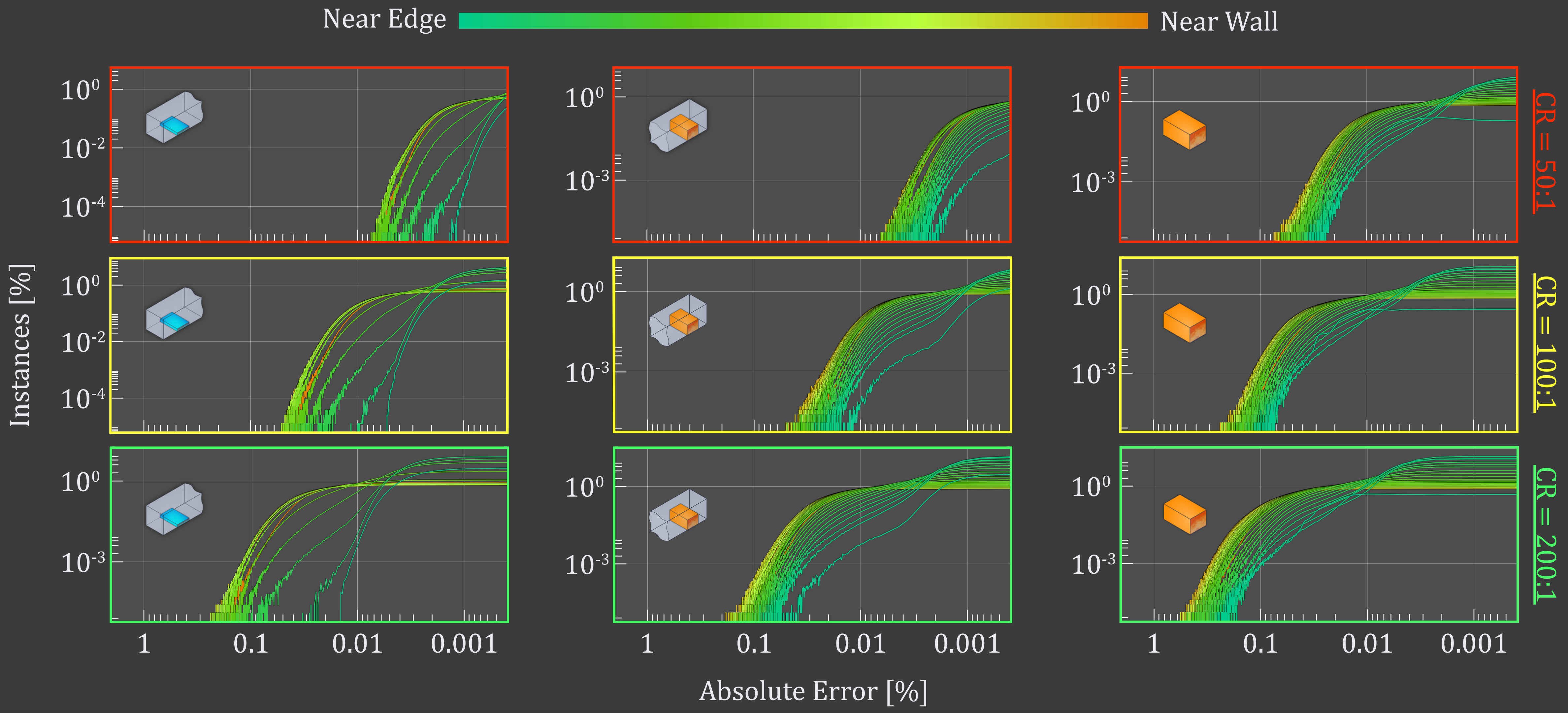


Figure 15 : Absolute error between original and reconstructed velocity field for compression ratios of 1 : 50, 1 : 100 and 1 : 200. The Left two columns were evaluated using two sub-domains at the in- and outlet of the complete simulation domain. For the right column, only the sub-domain at the outlet was compressed.

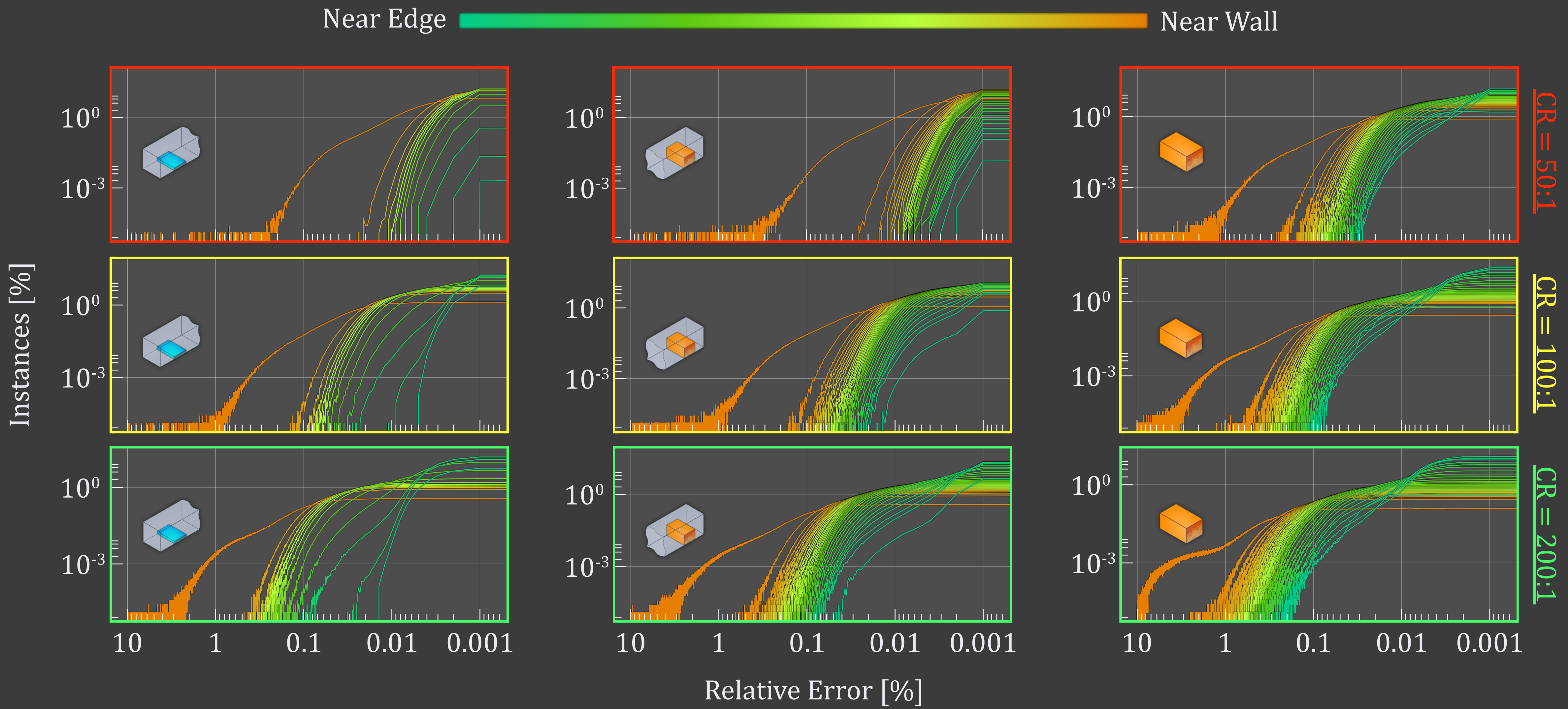
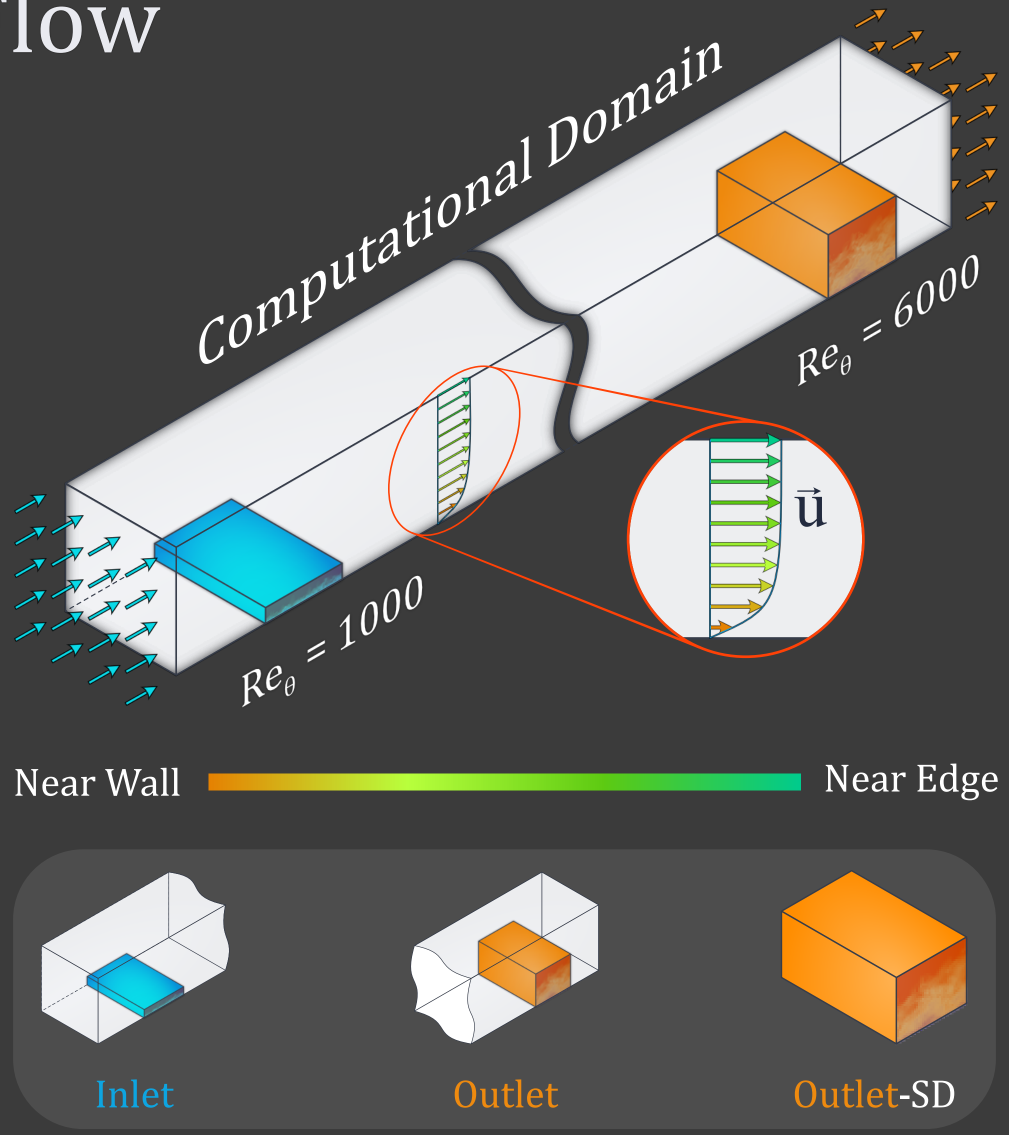
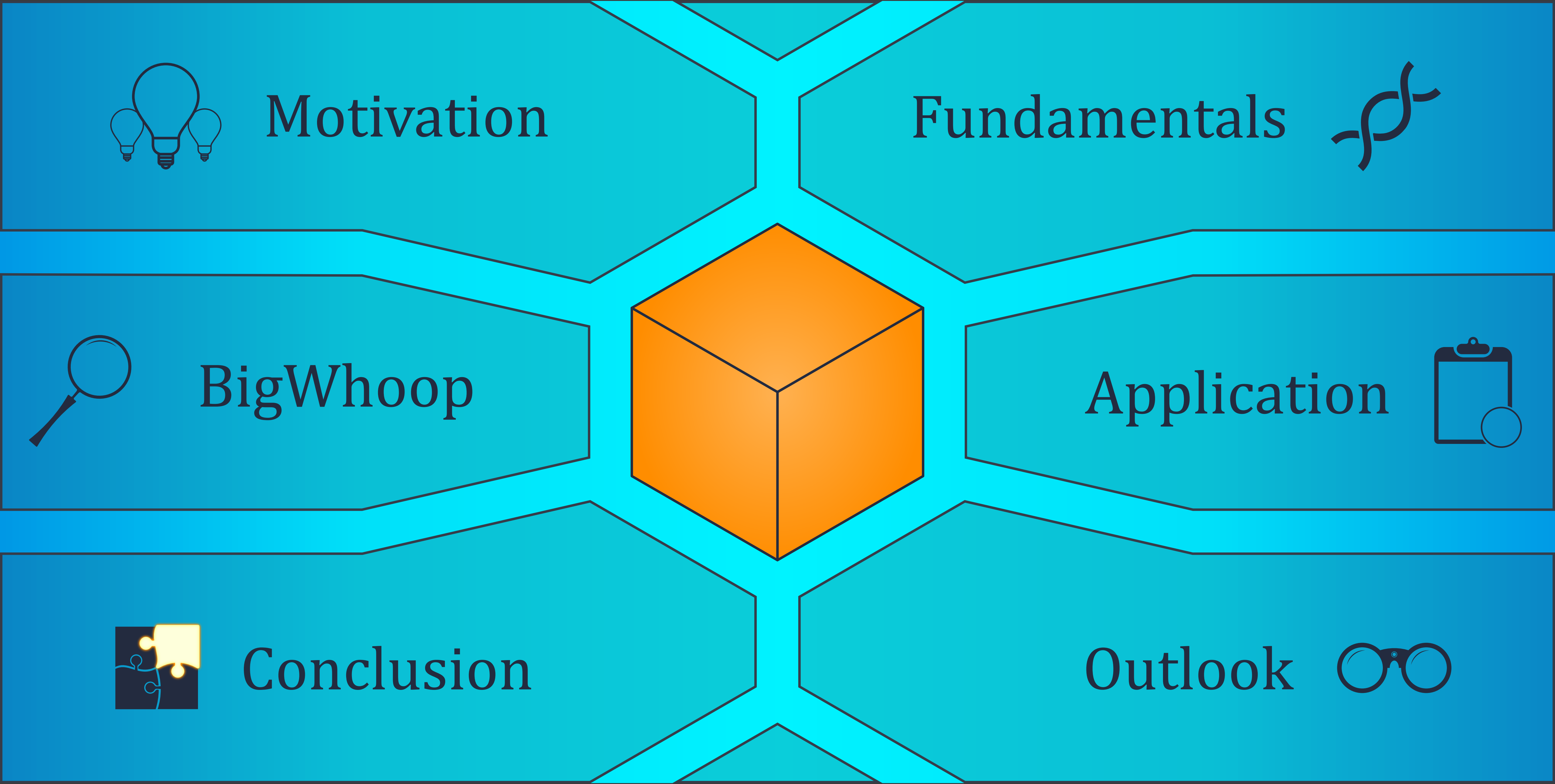


Figure 16 : Relative error between original and reconstructed velocity field for compression ratios of 1 : 50, 1 : 100 and 1 : 200. The Left two columns were evaluated using two sub-domains at the in- and outlet of the complete simulation domain. For the right column, only the sub-domain at the outlet was compressed.

Test Case: Turbulent Flat Plate Flow

- Compression library shows modest absolute error for high compression ratios.
- Separate sub-domain compression of region of interest doesn't offer improvements in reconstruction.
- Areas close to the wall exhibit significant relative errors due to values that are close or equal to zero.
- No-slip boundary condition can be lost due to the shift operation in the floating-to-fixed-point transformation.
- True zeroes should be encoded separately.




Motivation
Fundamentals
BigWhoop
Application
Conclusion
Outlook

Conclusion

- Compression can and should be used to effectively reduce the memory footprint of numerical data-sets.
- Data compression approach needs to fit the use case:
    - Fast dictionary encoders for lossless application (i.e. LZMA, Zstd).
    - Fast block transform coding for visualisation (i.e. ZFP).
    - Efficient discrete wavelet transform coding for data analysis (i.e. BigWhoop).
- Pre-compression truncation and parallelization paramount for good compression performance.

Motivation

Fundamentals

BigWhoop

Application

Conclusion

Outlook

Conclusion

- Global code-stream assembly.
- Adaption of BigWhoop codec for non-conforming compute grids.
- Decreased compression time using rate based sub-band skip.
- Implementation of streaming function for on the fly compression.